文章编号:1674-7070(2014)04-0289-17

丁锋[1,2,3]　汪菲菲[1]　汪学海[1]

多变量方程误差类系统的部分耦合迭代辨识方法

摘要

针对多变量方程误差滑动平均系统,利用最小二乘原理和迭代搜索原理,给出了增广随机梯度辨识方法、递推增广最小二乘辨识方法、梯度迭代辨识方法和最小二乘迭代辨识方法.针对多变量方程误差滑动平均系统和多变量方程误差自回归滑动平均系统,将多变量系统分解为一些子系统,利用耦合辨识概念,讨论了梯度迭代辨识方法、部分耦合(子系统)梯度迭代辨识方法、子系统最小二乘迭代方法和部分耦合子系统最小二乘迭代辨识方法.进一步结合数据滤波技术,研究了多变量方程误差自回归滑动平均系统的子系统梯度迭代辨识方法、部分耦合(子系统)梯度迭代辨识方法、部分耦合子系统最小二乘迭代辨识方法.文中给出了几个典型算法的计算步骤.

关键词

参数估计;迭代搜索原理;梯度搜索;最小二乘;数据滤波技术;辅助模型辨识思想;递阶辨识原理;耦合辨识概念;多变量系统

中图分类号 TP273

文献标志码 A

收稿日期 2014-07-24

资助项目 国家自然科学基金(61273194);江苏省自然科学基金(BK2012549);高等学校学科创新引智"111 计划"(B12018)

作者简介

丁锋,男,博士,教授,博士生导师,主要从事系统辨识、过程建模、自适应控制方面的研究.fding@ jiangnan.edu.cn

1 江南大学 物联网工程学院,无锡,214122

2 江南大学 控制科学与工程研究中心,无锡,214122

3 江南大学 教育部轻工过程先进控制重点实验室,无锡,214122

0 引言

迭代搜索辨识原理、辅助模型辨识思想、多新息辨识理论、递阶辨识原理、耦合辨识概念是重要的辨识理念和辨识方法研究思想[1-2],先后在《南京信息工程大学学报》连载论文中进行了讨论[3-13].从 2014 年开始继续刊登系统辨识连载论文,其中文献[14-16]将耦合辨识概念与多新息辨识理论相结合,研究了多元系统、多元伪线性回归系统和类多变量系统的耦合多新息辨识方法.本文将耦合辨识概念与迭代搜索辨识原理相结合,研究多变量方程误差类系统的辨识问题.

耦合辨识概念首先应用于辨识非均匀采样系统,提出了部分参数耦合随机梯度辨识方法[17],随后用于多元系统,提出了耦合最小二乘辨识方法[18],还被推广到研究各子系统间存在相同参数的多变量系统或多元系统的辨识问题[10].耦合辨识的研究对象可以是结构复杂的参数耦合线性和非线性多变量系统,能够减小参数耦合大规模多变量系统辨识算法的计算量.耦合辨识方法分为部分参数耦合辨识方法和全部参数耦合辨识方法,简称为部分耦合辨识方法和全耦合辨识方法.全耦合辨识方法有时也简称为耦合辨识方法.耦合辨识概念可以与辅助模型辨识思想、多新息辨识理论、递阶辨识原理、迭代搜索原理、牛顿搜索原理等相结合,来研究线性或非线性多变量系统的耦合辨识问题.

在实际应用中,人们只能采集到有限的一部分数据,而迭代辨识方法充分利用了采集到的数据,是一种离线辨识方法.迭代辨识方法可以用于辨识模型信息向量中含有未知项的系统辨识,其基本思想是利用批数据来刷新参数估计,信息向量中的未知项用前一步迭代参数估计进行估算,然后用估计的未知项代替信息向量中的真实未知项,参数估计利用代替后的信息向量进行刷新.这样便执行了一个交互估计和递阶辨识的过程.

滤波技术在目标跟踪、定位和参数估计方面有广泛的应用.通过使用有理多项式对系统的输入输出数据进行滤波,可以把要研究的系统分解成 2 个辨识模型,从而可以减少系统辨识模型参数向量的维数,达到提高辨识效率的目的.文献[19]研究了类多变量受控自回归滑动平均系统的基于滤波的最小二乘辨识算法,文献[20]利用关键项分离技术研究了 Hammerstein 系统的基于滤波的递推最小二乘辨识算法.

本文针对多变量方程误差类系统,讨论了梯度迭代算法和最小二乘迭代算法,进一步将多元系统分解成 m 个子系统(m 为输出数目),从其子系统梯度迭代辨识方法入手,推导出子系统梯度迭代辨识算法、部分耦合梯度迭代辨识算法和子系统最小二乘迭代辨识算法,并利用数据滤波技术,研究了多变量受控自回归自回归滑动平均系统的基于数据滤波的部分耦合子系统梯度迭代辨识算法和基于滤波的子系统最小二乘迭代辨识算法.文中给出了几个典型辨识算法的计算步骤,使读者更加清晰理解辨识方法的机理.

1 多变量受控自回归滑动平均系统

考虑多变量方程误差滑动平均系统,即多变量受控自回归滑动平均系统(多变量 CARMA 系统)[1-2]

$$\boldsymbol{A}(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)\boldsymbol{u}(t)+\boldsymbol{w}(t), \tag{1}$$

其中 $\boldsymbol{y}(t):=[y_1(t),y_2(t),\cdots,y_m(t)]^{\mathrm{T}}\in\mathbf{R}^m$ 为 m 维输出向量,$\boldsymbol{u}(t)\in\mathbf{R}^r$ 为 r 维输入向量,$\boldsymbol{w}(t)$ 为滑动平均过程,$\boldsymbol{A}(z)$ 和 $\boldsymbol{B}(z)$ 为单位后移算子 z^{-1} 的多项式矩阵:

$$\boldsymbol{A}(z):=\boldsymbol{I}+\boldsymbol{A}_1z^{-1}+\boldsymbol{A}_2z^{-2}+\cdots+\boldsymbol{A}_{n_a}z^{-n_a},\quad \boldsymbol{A}_i\in\mathbf{R}^{m\times m},$$

$$\boldsymbol{B}(z):=\boldsymbol{B}_1z^{-1}+\boldsymbol{B}_2z^{-2}+\cdots+\boldsymbol{B}_{n_b}z^{-n_b},\quad \boldsymbol{B}_i\in\mathbf{R}^{m\times r}.$$

对于多变量滑动平均过程,$\boldsymbol{w}(t)$ 可取 2 种形式:$\boldsymbol{w}(t)=\boldsymbol{D}(z)\boldsymbol{v}(t)$ 和 $\boldsymbol{w}(t)=d(z)\boldsymbol{v}(t)$,其中 $\boldsymbol{v}(t):=[v_1(t),v_2(t),\cdots,v_m(t)]^{\mathrm{T}}\in\mathbf{R}^m$ 为 m 维零均值白噪声向量,$\boldsymbol{D}(z)$ 为单位后移算子 z^{-1} 的多项式矩阵,$d(z)$ 为单位后移算子 z^{-1} 的首 1 多项式:

$$d(z):=1+d_1z^{-1}+d_2z^{-2}+\cdots+d_{n_d}z^{-n_d},\quad d_i\in\mathbf{R}.$$

本文考虑多项式情形,即所考虑的多变量系统如下:

$$\boldsymbol{A}(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)\boldsymbol{u}(t)+d(z)\boldsymbol{v}(t). \tag{2}$$

定义系统参数矩阵 $\boldsymbol{\theta}$,参数向量 $\boldsymbol{\alpha}$,信息向量 $\boldsymbol{\varphi}(t)$ 和信息矩阵 $\boldsymbol{\phi}(t)$ 如下:

$$\boldsymbol{\theta}^{\mathrm{T}}:=[\boldsymbol{A}_1,\boldsymbol{A}_2,\cdots,\boldsymbol{A}_{n_a},\boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_{n_b}]\in\mathbf{R}^{m\times n},$$

$$n:=mn_a+rn_b,$$

$$\boldsymbol{\alpha}:=[d_1,d_2,\cdots,d_{n_d}]^{\mathrm{T}}\in\mathbf{R}^{n_d},$$

$$\boldsymbol{\varphi}(t):=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}\in\mathbf{R}^n,$$

$$\boldsymbol{\phi}(t):=[\boldsymbol{v}(t-1),\boldsymbol{v}(t-2),\cdots,\boldsymbol{v}(t-n_d)]\in\mathbf{R}^{m\times n_d}.$$

于是,可以得到多变量 CARMA 系统(2)的递阶辨识模型:

$$\boldsymbol{y}(t)=\boldsymbol{\phi}(t)\boldsymbol{\alpha}+\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t)+\boldsymbol{v}(t). \tag{3}$$

递阶辨识模型(3)待辨识的参数既包含参数矩阵 $\boldsymbol{\theta}$,又包含参数向量 $\boldsymbol{\alpha}$,使得模型结构比较复杂.为了便于辨识,使用 Kronecker 积,将模型(3)中的信息向量 $\boldsymbol{\varphi}(t)$ 和信息矩阵 $\boldsymbol{\phi}(t)$ 化为一个大信息矩阵 $\boldsymbol{\Phi}(t)$,将参数矩阵 $\boldsymbol{\theta}$ 和参数向量 $\boldsymbol{\alpha}$ 化为一个大参数向量 $\boldsymbol{\vartheta}$,定义信息矩阵 $\boldsymbol{\Phi}(t)$ 和参数向量 $\boldsymbol{\vartheta}$ 如下:

$$\boldsymbol{\Phi}(t):=[\boldsymbol{\phi}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes\boldsymbol{I}_m]\in\mathbf{R}^{m\times n_0},\quad n_0:=n_d+mn,$$

$$\boldsymbol{\vartheta}:=\begin{bmatrix}\boldsymbol{\alpha}\\ \mathrm{col}[\boldsymbol{\theta}^{\mathrm{T}}]\end{bmatrix}\in\mathbf{R}^{n_0}.$$

式中⊗为 Kronecker 积符号,col[$\boldsymbol{X}$]定义为将矩阵 $\boldsymbol{X}$ 的列按次序排成的向量.于是递阶辨识模型(3)可以写为下列伪线性回归模型(pseudo-linear regressive model)[1-2]:

$$\boldsymbol{y}(t)=\boldsymbol{\Phi}(t)\boldsymbol{\vartheta}+\boldsymbol{v}(t). \tag{4}$$

新的参数向量 $\boldsymbol{\vartheta}$ 包含了系统的所有参数:不仅包含系统模型参数矩阵 $\boldsymbol{\theta}$,而且包含参数向量 $\boldsymbol{\alpha}$.假设 m,r,n_a,n_b 和 n_d 已知,且当 $t\leqslant 0$ 时,$\boldsymbol{y}(t)=0$,$\boldsymbol{u}(t)=0$,$\boldsymbol{v}(t)=0$.

这里的目标是基于最小二乘原理和迭代搜索原理,利用观测数据$\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,3,\cdots\}$研究这类多变量系统的增广随机梯度算法、递推增广最小二乘算法、梯度迭代算法和最小二乘迭代算法,来估计系统参数向量 $\boldsymbol{\vartheta}$.

1.1 增广随机梯度辨识算法

定义和极小化梯度准则函数

$$J_1(\boldsymbol{\vartheta}):=\|\boldsymbol{y}(t)-\boldsymbol{\Phi}(t)\boldsymbol{\vartheta}\|^2,$$

其中 $\|\boldsymbol{X}\|^2:=\mathrm{tr}[\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}]$ 定义为矩阵 $\boldsymbol{X}$ 的范数.令 $\hat{\boldsymbol{\vartheta}}(t)$ 为参数向量 $\boldsymbol{\vartheta}$ 在时刻 t 的估计.参考文献[1-2],利用负梯度搜索,可以得到估计参数向量 $\boldsymbol{\vartheta}$ 的随机梯度算法:

$$\hat{\boldsymbol{\vartheta}}(t)=\hat{\boldsymbol{\vartheta}}(t-1)-\frac{1}{2r(t)}\mathrm{grad}[J_1(\hat{\boldsymbol{\vartheta}}(t-1))]=$$

$$\hat{\boldsymbol{\vartheta}}(t-1)+\frac{\boldsymbol{\Phi}^{\mathrm{T}}(t)}{r(t)}[\boldsymbol{y}(t)-\boldsymbol{\Phi}(t)\hat{\boldsymbol{\vartheta}}(t-1)], \tag{5}$$

$$r(t)=r(t-1)+\|\boldsymbol{\Phi}(t)\|^2,\quad r(0)=1. \tag{6}$$

辨识的困难还在于 $\boldsymbol{\Phi}(t)$ 中包含了未知噪声项 $\boldsymbol{v}(t-j)$,使得算法(5)—(6)不可实现.解决的方法是将 $\boldsymbol{v}(t-j)$ 用其估计值 $\hat{\boldsymbol{v}}(t-j)$ 代替,代替后的信息矩阵 $\boldsymbol{\Phi}(t)$ 记作 $\hat{\boldsymbol{\Phi}}(t)$,即

$$\hat{\boldsymbol{\Phi}}(t):=[\hat{\boldsymbol{\phi}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes\boldsymbol{I}_m]\in\mathbf{R}^{m\times n_0},$$

$$\hat{\boldsymbol{\phi}}(t):=[\hat{\boldsymbol{v}}(t-1),\hat{\boldsymbol{v}}(t-2),\cdots,\hat{\boldsymbol{v}}(t-n_d)]\in\mathbf{R}^{m\times n_d},$$

$$\hat{\boldsymbol{v}}(t):=\boldsymbol{y}(t)-\hat{\boldsymbol{\Phi}}(t)\hat{\boldsymbol{\vartheta}}(t).$$

于是,就能够获得估计伪线性回归辨识模型(4)参数向量 $\boldsymbol{\vartheta}$ 的增广随机梯度算法(Extended

Stochastic Gradient algorithm,ESG 算法)[2]:

$$\hat{\boldsymbol{\vartheta}}(t)=\hat{\boldsymbol{\vartheta}}(t-1)+\frac{\hat{\boldsymbol{\Phi}}^{\mathrm{T}}(t)}{r(t)}[\boldsymbol{y}(t)-\hat{\boldsymbol{\Phi}}(t)\hat{\boldsymbol{\vartheta}}(t-1)],$$
$$\hat{\boldsymbol{\vartheta}}(0)=\mathbf{1}_{n_0}/p_0, \tag{7}$$
$$r(t)=r(t-1)+\|\hat{\boldsymbol{\Phi}}(t)\|^2,\quad r(0)=1, \tag{8}$$
$$\hat{\boldsymbol{\Phi}}(t)=[\hat{\boldsymbol{\phi}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes \boldsymbol{I}_m], \tag{9}$$
$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{10}$$
$$\hat{\boldsymbol{\phi}}(t)=[\hat{\boldsymbol{v}}(t-1),\hat{\boldsymbol{v}}(t-2),\cdots,\hat{\boldsymbol{v}}(t-n_d)], \tag{11}$$
$$\hat{\boldsymbol{v}}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\Phi}}(t)\hat{\boldsymbol{\vartheta}}(t). \tag{12}$$

ESG 算法(7)—(12)的计算步骤如下:

① 置初值:令 $t=1$, $\hat{\boldsymbol{\vartheta}}(0)=\mathbf{1}_{n_0}/p_0$, $r(0)=1$, $\hat{\boldsymbol{v}}(-j)=\mathbf{1}_m/p_0$, $j\geqslant 0$, $p_0=10^6$.

② 收集输入输出数据 $\boldsymbol{u}(t)$ 和 $\boldsymbol{y}(t)$,用式(10)和式(11)构成信息向量 $\boldsymbol{\varphi}(t)$ 和信息矩阵 $\hat{\boldsymbol{\phi}}(t)$,以构成式(9)中的 $\hat{\boldsymbol{\Phi}}(t)$.

③ 用式(8)计算 $r(t)$,用式(7)刷新参数估计 $\hat{\boldsymbol{\vartheta}}(t)$.

④ 用式(12)计算 $\hat{\boldsymbol{v}}(t)$.

⑤ t 增 1,转到第②步.

1.2 递推增广最小二乘辨识算法

定义和极小化最小二乘准则函数

$$J_2(\boldsymbol{\vartheta}):=\sum_{j=1}^{t}\|\boldsymbol{y}(j)-\boldsymbol{\Phi}(j)\boldsymbol{\vartheta}\|^2.$$

参考文献[2]中的推导方法,可以得到估计参数向量 $\boldsymbol{\vartheta}$ 的递推最小二乘算法:

$$\hat{\boldsymbol{\vartheta}}(t)=\hat{\boldsymbol{\vartheta}}(t-1)+\boldsymbol{P}(t)\boldsymbol{\Phi}^{\mathrm{T}}(t)[\boldsymbol{y}(t)-\boldsymbol{\Phi}(t)\hat{\boldsymbol{\vartheta}}(t-1)], \tag{13}$$
$$\boldsymbol{P}^{-1}(t)=\boldsymbol{P}^{-1}(t-1)+\boldsymbol{\Phi}^{\mathrm{T}}(t)\boldsymbol{\Phi}(t). \tag{14}$$

为避免计算式(14)中大协方差阵 $\boldsymbol{P}(t)\in\mathbf{R}^{n_0\times n_0}$ 的逆,应用矩阵求逆公式

$$(\boldsymbol{A}+\boldsymbol{BC})^{-1}=\boldsymbol{A}^{-1}-\boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{I}+\boldsymbol{CA}^{-1}\boldsymbol{B})^{-1}\boldsymbol{CA}^{-1}$$

于式(14),并引入增益向量 $\boldsymbol{L}(t):=\boldsymbol{P}(t)\boldsymbol{\Phi}^{\mathrm{T}}(t)\in\mathbf{R}^{n_0\times m}$,于是得到等价的递推最小二乘算法:

$$\hat{\boldsymbol{\vartheta}}(t)=\hat{\boldsymbol{\vartheta}}(t-1)+\boldsymbol{L}(t)[\boldsymbol{y}(t)-\boldsymbol{\Phi}(t)\hat{\boldsymbol{\vartheta}}(t-1)], \tag{15}$$
$$\boldsymbol{L}(t)=\boldsymbol{P}(t-1)\boldsymbol{\Phi}^{\mathrm{T}}(t)[\boldsymbol{I}_m+\boldsymbol{\Phi}(t)\boldsymbol{P}(t-1)\boldsymbol{\Phi}^{\mathrm{T}}(t)]^{-1}, \tag{16}$$
$$\boldsymbol{P}(t)=\boldsymbol{P}(t-1)-\boldsymbol{L}(t)\boldsymbol{\Phi}(t)\boldsymbol{P}(t-1). \tag{17}$$

为了实现参数估计,将 $\boldsymbol{\Phi}(t)$ 中包含的未知回归项 $\boldsymbol{v}(t-j)$ 用其估计值 $\hat{\boldsymbol{v}}(t-j)$ 代替,并将代替后的 $\boldsymbol{\Phi}(t)$ 记作 $\hat{\boldsymbol{\Phi}}(t)$,则得到可实现的估计参数向量 $\boldsymbol{\vartheta}$ 的递推增广最小二乘算法(Recursive Extended Least Squares algorithm,RELS 算法):

$$\hat{\boldsymbol{\vartheta}}(t)=\hat{\boldsymbol{\vartheta}}(t-1)+\boldsymbol{L}(t)[\boldsymbol{y}(t)-\hat{\boldsymbol{\Phi}}(t)\hat{\boldsymbol{\vartheta}}(t-1)],$$
$$\hat{\boldsymbol{\vartheta}}(0)=\mathbf{1}_{n_0}/p_0, \tag{18}$$
$$\boldsymbol{L}(t)=\boldsymbol{P}(t-1)\hat{\boldsymbol{\Phi}}^{\mathrm{T}}(t)[\boldsymbol{I}+\hat{\boldsymbol{\Phi}}(t)\boldsymbol{P}(t-1)\hat{\boldsymbol{\Phi}}^{\mathrm{T}}(t)]^{-1}, \tag{19}$$
$$\boldsymbol{P}(t)=\boldsymbol{P}(t-1)-\boldsymbol{L}(t)\hat{\boldsymbol{\Phi}}(t)\boldsymbol{P}(t-1),\quad \boldsymbol{P}(0)=p_0\boldsymbol{I}_{n_0}, \tag{20}$$
$$\hat{\boldsymbol{\Phi}}(t)=[\hat{\boldsymbol{\phi}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes \boldsymbol{I}_m], \tag{21}$$
$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{22}$$
$$\hat{\boldsymbol{\phi}}(t)=[\hat{\boldsymbol{v}}(t-1),\hat{\boldsymbol{v}}(t-2),\cdots,\hat{\boldsymbol{v}}(t-n_d)], \tag{23}$$
$$\hat{\boldsymbol{v}}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\Phi}}(t)\hat{\boldsymbol{\vartheta}}(t). \tag{24}$$

RELS 算法(18)—(24)的计算步骤如下:

① 置初值:令 $t=1$, $\hat{\boldsymbol{\vartheta}}(0)=\mathbf{1}_{n_0}/p_0$, $\boldsymbol{P}(0)=p_0\boldsymbol{I}_{n_0}$, $\hat{\boldsymbol{v}}(-j)=\mathbf{1}_m/p_0$, $j\geqslant 0$, $p_0=10^6$.

② 收集输入输出数据 $\boldsymbol{u}(t)$ 和 $\boldsymbol{y}(t)$,用式(22)和式(23)构成信息向量 $\boldsymbol{\varphi}(t)$ 和信息矩阵 $\hat{\boldsymbol{\phi}}(t)$,以构成式(21)中的 $\hat{\boldsymbol{\Phi}}(t)$.

③ 用式(19)和(20)分别计算 $\boldsymbol{L}(t)$ 和 $\boldsymbol{P}(t)$,用式(18)刷新参数估计 $\hat{\boldsymbol{\vartheta}}(t)$.

④ 用式(24)计算 $\hat{\boldsymbol{v}}(t)$.

⑤ t 增 1,转到第②步,继续递推计算.

从以上递推计算过程来看,如果收集的数据足够长,RELS 算法就能够收敛到真实值.但是在实际工程应用中,只可能采集到有限的一部分数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,\cdots,L\}$($L$ 为数据长度),而 RELS 算法在第 t 次递推计算参数估计中,只使用了 t 时刻以及 t 时刻以前的数据,其余的从 t 以后的数据都被忽视了.虽然在整个 RELS 算法计算过程中,利用了系统的所有数据,但是并没有充分利用这些数据.下面的迭代辨识方法,在每次迭代计算过程中,都充分利用了所采集到的数据.

1.3 梯度迭代辨识算法

下面研究多变量 CARMA 系统的梯度迭代辨识方法.

考虑从 $i=t-p+1$ 到 $i=t$ 的最新 p 组数据,定义堆积输出向量 $\boldsymbol{Y}(t)$,堆积信息矩阵 $\boldsymbol{\Phi}(t)$,堆积白噪声向量 $\boldsymbol{V}(t)$ 如下:

$$\boldsymbol{Y}(t):=\begin{bmatrix}\boldsymbol{y}(t)\\ \boldsymbol{y}(t-1)\\ \vdots\\ \boldsymbol{y}(t-p+1)\end{bmatrix}\in\mathbf{R}^{mp},$$

$$\boldsymbol{\Gamma}(t):=\begin{bmatrix}\boldsymbol{\Phi}(t)\\ \boldsymbol{\Phi}(t-1)\\ \vdots\\ \boldsymbol{\Phi}(t-p+1)\end{bmatrix}\in\mathbf{R}^{(mp)\times n_0}.$$

令 $k=1,2,3,\cdots$ 为迭代变量,$\hat{\boldsymbol{\vartheta}}_k(t)$ 为 $\boldsymbol{\vartheta}$ 在第 k 次迭代的参数估计,$\mu_k(t)\geqslant 0$ 为时变步长(时变收敛因子).使用负梯度搜索,定义并极小化准则函数

$$J_3(\boldsymbol{\vartheta}):=\|\boldsymbol{Y}(t)-\boldsymbol{\Gamma}(t)\boldsymbol{\vartheta}\|^2,$$

得到梯度迭代算法:

$$\hat{\boldsymbol{\vartheta}}_k(t)=\hat{\boldsymbol{\vartheta}}_{k-1}(t)-\frac{\mu_k(t)}{2}\mathrm{grad}[J_3(\hat{\boldsymbol{\vartheta}}_{k-1}(t))]=$$
$$\hat{\boldsymbol{\vartheta}}_{k-1}(t)+\mu_k(t)\boldsymbol{\Gamma}^{\mathrm{T}}(t)[\boldsymbol{Y}(t)-\boldsymbol{\Gamma}(t)\hat{\boldsymbol{\vartheta}}_{k-1}(t)]. \tag{25}$$

由于 $\boldsymbol{\Gamma}(t)$ 中包含了不可测噪声项 $\boldsymbol{v}(t-j)$,式(25)计算估计 $\hat{\boldsymbol{\vartheta}}(t)$ 不可能实现.采用递阶辨识原理,信息矩阵 $\boldsymbol{\phi}(t)$ 中包含的不可测噪声项 $\boldsymbol{v}(t-j)$ 用其第 $k-1$ 次迭代估计值 $\hat{\boldsymbol{v}}_{k-1}(t-j)$ 代替,代替后的 $\boldsymbol{\phi}(t)$ 记作 $\hat{\boldsymbol{\phi}}_k(t)$,代替后的 $\boldsymbol{\Phi}(t)$ 记作 $\hat{\boldsymbol{\Phi}}_k(t)$,代替后的 $\boldsymbol{\Gamma}(t)$ 记作 $\hat{\boldsymbol{\Gamma}}_k(t)$.定义

$$\hat{\boldsymbol{\Gamma}}_k(t):=\begin{bmatrix}\hat{\boldsymbol{\Phi}}_k(t)\\ \hat{\boldsymbol{\Phi}}_k(t-1)\\ \vdots\\ \hat{\boldsymbol{\Phi}}_k(t-p+1)\end{bmatrix}\in\mathbf{R}^{(mp)\times n_0},$$

$$\hat{\boldsymbol{\Phi}}_k(t):=[\hat{\boldsymbol{\phi}}_k(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes\boldsymbol{I}_m]\in\mathbf{R}^{m\times n_0},$$

$$\hat{\boldsymbol{\phi}}_k(t):=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\in\mathbf{R}^{m\times n_d}.$$

用 $\hat{\boldsymbol{\Gamma}}_k(t)$ 代替式(25)中 $\boldsymbol{\Gamma}(t)$,可得到多变量 CARMA 系统的梯度迭代算法(Gradient based Iterative algorithm,GI 算法):

$$\hat{\boldsymbol{\vartheta}}_k(t)=\hat{\boldsymbol{\vartheta}}_{k-1}(t)+\mu_k(t)\hat{\boldsymbol{\Gamma}}_k^{\mathrm{T}}(t)[\boldsymbol{Y}(t)-\hat{\boldsymbol{\Gamma}}_k(t)\hat{\boldsymbol{\vartheta}}_{k-1}(t)],$$
$$k=1,2,3,\cdots, \tag{26}$$

$$\hat{\boldsymbol{\Gamma}}_k(t)=[\hat{\boldsymbol{\Phi}}_k^{\mathrm{T}}(t),\hat{\boldsymbol{\Phi}}_k^{\mathrm{T}}(t-1),\cdots,\hat{\boldsymbol{\Phi}}_k^{\mathrm{T}}(t-p+1)]^{\mathrm{T}}, \tag{27}$$

$$\boldsymbol{Y}(t)=[\boldsymbol{y}^{\mathrm{T}}(t),\boldsymbol{y}^{\mathrm{T}}(t-1),\cdots,\boldsymbol{y}^{\mathrm{T}}(t-p+1)]^{\mathrm{T}}, \tag{28}$$

$$\hat{\boldsymbol{\Phi}}_k(t)=[\hat{\boldsymbol{\phi}}_k(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes\boldsymbol{I}_m], \tag{29}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{30}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)], \tag{31}$$

$$\hat{\boldsymbol{v}}_k(t-j)=\boldsymbol{y}(t-j)-\hat{\boldsymbol{\Phi}}_k(t-j)\hat{\boldsymbol{\vartheta}}_k(t),\quad j=0,1,\cdots,n_d, \tag{32}$$

$$\mu_k(t)\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_k^{\mathrm{T}}(t)\hat{\boldsymbol{\Gamma}}_k(t)]. \tag{33}$$

GI 算法(26)—(33)的计算步骤如下:

① 确定 p,令 $t=p(p\gg n_0)$.收集输入输出数据 $\{\boldsymbol{u}(j),\boldsymbol{y}(j):j=1,2,\cdots,p-1\}$,给定参数估计精度 ε.

② 令 $k=1$,$\hat{\boldsymbol{\vartheta}}_0(t)=\mathbf{1}_{n_0}/p_0$,$\hat{\boldsymbol{v}}_0(t-j)=\mathbf{1}_m/p_0(j=1,2,\cdots,n_d)$,$p_0=10^6$.

③ 收集输入输出数据 $\boldsymbol{u}(t)$ 和 $\boldsymbol{y}(t)$,用式(28)构造 $\boldsymbol{Y}(t)$,用式(30)构造 $\boldsymbol{\varphi}(t)$.

④ 用式(31)构造 $\hat{\boldsymbol{\phi}}_k(t)$,构成式(29)中的 $\hat{\boldsymbol{\Phi}}_k(t)$,用式(27)构造 $\hat{\boldsymbol{\Gamma}}_k(t)$.

⑤ 根据式(33)选择 $\mu_k(t)=2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_k^{\mathrm{T}}(t)\cdot\hat{\boldsymbol{\Gamma}}_k(t)]$,用式(26)刷新参数估计 $\hat{\boldsymbol{\vartheta}}_k(t)$.

⑥ 用式(32)计算 $\hat{\boldsymbol{v}}_k(t-j)$.

⑦ 比较 $\hat{\boldsymbol{\vartheta}}_k(t)$ 和 $\hat{\boldsymbol{\vartheta}}_{k-1}(t)$,如果 $\|\hat{\boldsymbol{\vartheta}}_k(t)-\hat{\boldsymbol{\vartheta}}_{k-1}(t)\|\geqslant\varepsilon$,$k$ 增 1,转到步骤④;否则获得迭代次数 k 和参数估计 $\hat{\boldsymbol{\vartheta}}_k(t)$,置 $\hat{\boldsymbol{\vartheta}}_0(t+1):=\hat{\boldsymbol{\vartheta}}_k(t)$,$t:=t+1$,转到步骤③.

1.4 最小二乘迭代辨识算法

定义并极小化准则函数

$$J_4(\boldsymbol{\vartheta}):=\|\boldsymbol{Y}(t)-\boldsymbol{\Gamma}(t)\boldsymbol{\vartheta}\|^2,$$

得到参数向量 $\boldsymbol{\vartheta}$ 的最小二乘估计:

$$\hat{\boldsymbol{\vartheta}}(t)=[\boldsymbol{\Gamma}^{\mathrm{T}}(t)\boldsymbol{\Gamma}(t)]^{-1}\boldsymbol{\Gamma}^{\mathrm{T}}(t)\boldsymbol{Y}(t). \tag{34}$$

辨识的困难在于 $\boldsymbol{\Gamma}(t)$ 中包含了不可测噪声项 $\boldsymbol{v}(t-j)$,算法(34)计算参数估计 $\hat{\boldsymbol{\vartheta}}(t)$ 无法实现.同理,用 $\hat{\boldsymbol{\Gamma}}_k(t)$ 代替 $\boldsymbol{\Gamma}(t)$,能够得到辨识多变量 CARMA 系统参数向量 $\boldsymbol{\vartheta}$ 的最小二乘迭代算法(Least Squares based Iterative algorithm,LSI 算法):

$$\hat{\boldsymbol{\vartheta}}_k(t)=[\hat{\boldsymbol{\Gamma}}_k^{\mathrm{T}}(t)\hat{\boldsymbol{\Gamma}}_k(t)]^{-1}\hat{\boldsymbol{\Gamma}}_k^{\mathrm{T}}(t)\boldsymbol{Y}(t),k=1,2,3,\cdots, \tag{35}$$

$$\hat{\boldsymbol{\Gamma}}_k(t)=[\hat{\boldsymbol{\Phi}}_k^{\mathrm{T}}(t),\hat{\boldsymbol{\Phi}}_k^{\mathrm{T}}(t-1),\cdots,\hat{\boldsymbol{\Phi}}_k^{\mathrm{T}}(t-p+1)]^{\mathrm{T}}, \tag{36}$$

$$\boldsymbol{Y}(t)=[\boldsymbol{y}^{\mathrm{T}}(t),\boldsymbol{y}^{\mathrm{T}}(t-1),\cdots,\boldsymbol{y}^{\mathrm{T}}(t-p+1)]^{\mathrm{T}}, \tag{37}$$

$$\hat{\boldsymbol{\Phi}}_k(t)=[\hat{\boldsymbol{\phi}}_k(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)\otimes\boldsymbol{I}_m], \tag{38}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{39}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)], \tag{40}$$

$$\hat{\boldsymbol{v}}_k(t-j)=\boldsymbol{y}(t-j)-\hat{\boldsymbol{\Phi}}_k(t-j)\hat{\boldsymbol{\vartheta}}_k(t),\quad j=0,1,\cdots,n_d. \tag{41}$$

LSI 算法(35)—(41)的计算步骤如下:

① 确定 p,令 $t=p(p\gg n_0)$.收集输入输出数据 $\{\boldsymbol{u}(j),\boldsymbol{y}(j):j=1,2,\cdots,p-1\}$,给定参数估计精度 ε.

② 令 $k=1$,$\hat{\boldsymbol{v}}_0(t)=$ 随机数(这样设置初值是为了保证式(35)中矩阵可逆).

③ 收集输入输出数据 $\boldsymbol{u}(t)$ 和 $\boldsymbol{y}(t)$,用式(37)构造 $\boldsymbol{Y}(t)$,用式(39)构造 $\boldsymbol{\varphi}(t)$.

④ 用式(40)构造 $\hat{\boldsymbol{\phi}}_k(t)$,构成式(38)中的 $\hat{\boldsymbol{\Phi}}_k(t)$,用式(36)构造 $\hat{\boldsymbol{\Gamma}}_k(t)$.

⑤ 用式(35)刷新参数估计 $\hat{\boldsymbol{\vartheta}}_k(t)$ 的值.

⑥ 用式(41)计算 $\hat{\boldsymbol{v}}_k(t-j)$.

⑦ 比较 $\hat{\boldsymbol{\vartheta}}_k(t)$ 和 $\hat{\boldsymbol{\vartheta}}_{k-1}(t)$,如果 $\|\hat{\boldsymbol{\vartheta}}_k(t)-\hat{\boldsymbol{\vartheta}}_{k-1}(t)\|>\varepsilon$,$k$ 增 1,转到步骤④;否则获得迭代次数 k 和参数估计 $\hat{\boldsymbol{\vartheta}}_k(t)$,$t$ 增 1,转到步骤③.

GI 算法和 LSI 算法在每一次迭代计算参数估计

时，使用了长度 p 的数据窗里的输入输出数据，故具有跟踪时变参数的能力. 迭代算法一般用于信息向量中含有未知项系统的辨识问题.

2 多变量 CARMA 系统的迭代辨识方法

对于多变量系统，如果辨识模型的每个子系统包含一个共同的参数向量和一个相同的信息向量，每个子系统包含一个不同的信息向量和一个不同的参数向量，如式(3)和式(44). 为减小辨识算法的计算量，可以利用递阶辨识原理，推导出递阶随机梯度辨识方法、递阶最小二乘辨识方法、递阶梯度迭代辨识方法、递阶最小二乘迭代辨识方法[21-22]；也可利用耦合辨识概念，研究和提出耦合随机梯度辨识方法、耦合最小二乘辨识方法、耦合多新息随机梯度辨识方法、耦合多新息最小二乘辨识方法等[16]. 下面利用耦合辨识概念，研究多变量 CARMA 系统的部分耦合子系统梯度迭代辨识算法和子系统最小二乘迭代辨识算法.

2.1 子系统辨识模型

考虑多变量 CARMA 系统(2)的递阶辨识模型(3)，重写如下：

$$\boldsymbol{y}(t)=\boldsymbol{\phi}(t)\boldsymbol{\alpha}+\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t)+\boldsymbol{v}(t), \tag{42}$$

其中系统参数矩阵 $\boldsymbol{\theta}$，参数向量 $\boldsymbol{\alpha}$，信息向量 $\boldsymbol{\varphi}(t)$ 和信息矩阵 $\boldsymbol{\phi}(t)$ 的定义同上.

令 $\boldsymbol{\phi}_i^{\mathrm{T}}(t)$ 为信息矩阵 $\boldsymbol{\phi}(t)$ 的第 i 行，$\boldsymbol{\theta}_i$ 为参数矩阵 $\boldsymbol{\theta}$ 的第 i 列，即

$$\boldsymbol{\phi}(t):=[\boldsymbol{\phi}_1(t),\boldsymbol{\phi}_2(t),\cdots,\boldsymbol{\phi}_m(t)]^{\mathrm{T}}\in\mathbf{R}^{m\times n},$$
$$\boldsymbol{\theta}:=[\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_m]\in\mathbf{R}^{(nr)\times m}.$$

将递阶辨识模型(42)进一步写为

$$\begin{bmatrix}y_1(t)\\y_2(t)\\\vdots\\y_m(t)\end{bmatrix}=\begin{bmatrix}\boldsymbol{\phi}_1^{\mathrm{T}}(t)\\\boldsymbol{\phi}_2^{\mathrm{T}}(t)\\\vdots\\\boldsymbol{\phi}_m^{\mathrm{T}}(t)\end{bmatrix}\boldsymbol{\alpha}+\begin{bmatrix}\boldsymbol{\theta}_1^{\mathrm{T}}\\\boldsymbol{\theta}_2^{\mathrm{T}}\\\vdots\\\boldsymbol{\theta}_m^{\mathrm{T}}\end{bmatrix}\boldsymbol{\varphi}(t)+\begin{bmatrix}v_1(t)\\v_2(t)\\\vdots\\v_m(t)\end{bmatrix}, \tag{43}$$

则式(43)可以分解为 m 个子辨识模型

$$\begin{aligned}y_i(t)&=\boldsymbol{\phi}_i^{\mathrm{T}}(t)\boldsymbol{\alpha}+\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\varphi}(t)+v_i(t)\\&=\boldsymbol{\phi}_i^{\mathrm{T}}(t)\boldsymbol{\alpha}+\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\theta}_i+v_i(t) \qquad (44)\\&=[\boldsymbol{\phi}_i^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]\begin{bmatrix}\boldsymbol{\alpha}\\\boldsymbol{\theta}_i\end{bmatrix}+v_i(t),\quad i=1,2,\cdots,m. \qquad (45)\end{aligned}$$

定义子系统信息向量为 $\boldsymbol{\psi}_i(t):=\begin{bmatrix}\boldsymbol{\phi}_i(t)\\\boldsymbol{\varphi}(t)\end{bmatrix}\in\mathbf{R}^{n+nr}$，则子辨识模型(44)可以写成

$$y_i(t)=\boldsymbol{\psi}_i^{\mathrm{T}}(t)\begin{bmatrix}\boldsymbol{\alpha}\\\boldsymbol{\theta}_i\end{bmatrix}+v_i(t),\quad i=1,2,\cdots,m. \tag{46}$$

在式(44)的 m 个子辨识模型中，每个子辨识模型有一个共同的子参数向量 $\boldsymbol{\alpha}$ 和一个共同的子信息向量 $\boldsymbol{\varphi}(t)$，每个子辨识模型还包含了一个不同的子参数向量 $\boldsymbol{\theta}_i$ 和一个不同的子信息向量 $\boldsymbol{\phi}_i(t)$. 下面研究子辨识模型(46)的迭代辨识方法.

2.2 子系统梯度迭代辨识算法

考虑数据窗长度为 L 的有限量测数据，即从 $t=1$ 到 $t=L$ 的一组数据，定义子系统堆积输出向量 $\boldsymbol{Y}_i(L)$，堆积信息矩阵 $\boldsymbol{\Psi}_i(L)$，堆积白噪声向量 $\boldsymbol{V}_i(L)$ 如下：

$$\boldsymbol{Y}_i(L):=\begin{bmatrix}y_i(1)\\y_i(2)\\\vdots\\y_i(L)\end{bmatrix}\in\mathbf{R}^{L},\quad \boldsymbol{\Gamma}_i(L):=\begin{bmatrix}\boldsymbol{\psi}_i^{\mathrm{T}}(1)\\\boldsymbol{\psi}_i^{\mathrm{T}}(2)\\\vdots\\\boldsymbol{\psi}_i^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times(n+nr)},$$

$$\boldsymbol{V}_i(L):=\begin{bmatrix}v_i(1)\\v_i(2)\\\vdots\\v_i(L)\end{bmatrix}\in\mathbf{R}^{L}.$$

$\boldsymbol{Y}_i(L)$ 和 $\boldsymbol{\Gamma}_i(L)$ 中包含了所有量测输入输出数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,\cdots,L\}$. 由式(46)可得

$$\boldsymbol{Y}_i(L)=\boldsymbol{\Gamma}_i(L)\begin{bmatrix}\boldsymbol{\alpha}\\\boldsymbol{\theta}_i\end{bmatrix}+\boldsymbol{V}_i(L),\quad i=1,2,\cdots,m. \tag{47}$$

令 $\mu_k\geqslant 0$ 为步长，$\hat{\boldsymbol{\alpha}}_k$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$ 分别为 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\theta}_i$ 在第 k 次迭代的参数估计. 使用负梯度搜索，定义并极小化准则函数

$$J_5(\boldsymbol{\alpha},\boldsymbol{\theta}_i):=\left\{\boldsymbol{Y}_i(L)-\boldsymbol{\Gamma}_i(L)\begin{bmatrix}\boldsymbol{\alpha}\\\boldsymbol{\theta}_i\end{bmatrix}\right\}^2,$$

得到估计参数向量 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\theta}_i$ 的迭代算法：

$$\begin{aligned}\begin{bmatrix}\hat{\boldsymbol{\alpha}}_k\\\hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}&=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\\hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}-\frac{\mu_k}{2}\mathrm{grad}[J_5(\hat{\boldsymbol{\alpha}}_{k-1},\hat{\boldsymbol{\theta}}_{i,k-1})]=\\&\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\\hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k\boldsymbol{\Gamma}_i^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_i(L)-\boldsymbol{\Gamma}_i(L)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\\hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\}. \qquad (48)\end{aligned}$$

由于 $\boldsymbol{\Gamma}_i(L)$ 中包含了不可测噪声项 $\boldsymbol{v}(t-j)$，式(48)计算最小二乘估计 $\hat{\boldsymbol{\alpha}}_k$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$ 不可能实现. 采用递阶辨识原理，将未知变量 $\boldsymbol{\Gamma}_i(L)$ 中 $\boldsymbol{\phi}_i(t)$ 包含的不可测噪声项 $\boldsymbol{v}(t-j)$ 用第 $k-1$ 次的迭代估计 $\hat{\boldsymbol{v}}_{k-1}(t-j)$ 代替，定义 $\boldsymbol{\Gamma}_i(L)$ 和 $\boldsymbol{\phi}_i(t)$ 的第 k 次迭代估计：

$$\hat{\boldsymbol{\Gamma}}_{i,k}(L):=\begin{bmatrix}\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(1)\\\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(2)\\\vdots\\\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times(n+nr)},$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t):=\begin{bmatrix}\hat{\boldsymbol{\phi}}_{i,k}(t)\\ \boldsymbol{\varphi}(t)\end{bmatrix}\in\mathbf{R}^{m\times(n+nr)},$$

$$\hat{\boldsymbol{\phi}}_k(t):=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]=$$
$$[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}}\in\mathbf{R}^{m\times n_d},$$

用 $\hat{\boldsymbol{\Gamma}}_{i,k}(L)$ 代替式(48)中 $\boldsymbol{\Gamma}_i(L)$,能够得到多变量CARMA系统的子系统梯度迭代算法:

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_k\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_i(L)-\hat{\boldsymbol{\Gamma}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},$$
$$i=1,2,\cdots,m,$$
$$\hat{\boldsymbol{\Gamma}}_{i,k}(L):=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},$$
$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},$$
$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},$$
$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},$$
$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]=$$
$$[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},$$
$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},$$
$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix},$$
$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)].$$

为区分每个子系统中的估计 $\hat{\boldsymbol{\alpha}}_k$,给它加下标 i,记作 $\hat{\boldsymbol{\alpha}}_{i,k}$,便得到下列子系统梯度迭代算法(Subsystem Gradient based Iterative algorithm, S-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_i(L)-\hat{\boldsymbol{\Gamma}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\hat{\boldsymbol{\alpha}}\end{bmatrix}\right\},\quad(49)$$
$$\hat{\boldsymbol{\Gamma}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\quad(50)$$
$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\quad(51)$$
$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\quad(52)$$
$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\quad(53)$$
$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\quad(54)$$
$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\quad(55)$$
$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},\quad(56)$$
$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix},\quad(57)$$
$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)].\quad(58)$$

2.3 部分耦合子系统梯度迭代辨识算法

在上述S-GI算法(49)—(58)中,对参数向量 $\hat{\boldsymbol{\alpha}}_k$ 进行了 m 次估计,得到了 m 个 $\hat{\boldsymbol{\alpha}}_{i,k}$,而实际只需要一个估计,取每个子系统的估计 $\hat{\boldsymbol{\alpha}}_i(t)$ 的平均值作为参数向量 $\boldsymbol{\alpha}$ 的估计,即

$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m}\in\mathbf{R}^{n_d}.$$

用平均值 $\hat{\boldsymbol{\alpha}}_{k-1}$ 代替S-GI算法(49)—(58)中的 $\hat{\boldsymbol{\alpha}}_{i,k-1}$,就得到了一个简单的部分耦合子系统梯度迭代算法(Partially Coupled Subsystem Gradient based Iterative algorithm, PC-S-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k(t)\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_i(L)-\hat{\boldsymbol{\Gamma}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},\quad(59)$$
$$\hat{\boldsymbol{\Gamma}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\quad(60)$$
$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\quad(61)$$
$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\quad(62)$$
$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$
$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\quad(63)$$
$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\quad(64)$$
$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\quad(65)$$
$$\hat{\boldsymbol{v}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{\alpha}}_k-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\quad(66)$$
$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m},\quad(67)$$
$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}],\quad(68)$$
$$\mu_k(t)\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)].\quad(69)$$

2.4 部分耦合梯度迭代辨识算法

对于迭代参数估计算法,参数估计随着迭代变量 k 的增大而收敛于真参数,故可以认为第 $i-1$ 个子系统在时刻 k 的参数估计 $\hat{\boldsymbol{\alpha}}_{i-1,k}$ 比第 i 个子系统在时刻 $k-1$ 的参数估计 $\hat{\boldsymbol{\alpha}}_{i,k-1}$ 更接近真参数 $\boldsymbol{\alpha}$.参考文献[10,14-15,17]的耦合辨识思想,用 $\hat{\boldsymbol{\alpha}}_{m,k-1}$ 代替式(49)中 $i=1$ 时的 $\hat{\boldsymbol{\alpha}}_{1,k-1}$,用 $\hat{\boldsymbol{\alpha}}_{i-1,k}$ 代替式(49)右边的 $\hat{\boldsymbol{\alpha}}_{i,k-1}$,则得到部分耦合梯度迭代算法(Partially Coupled Gradient based Iterative algorithm, PC-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{1,k}\\ \hat{\boldsymbol{\theta}}_{1,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{m,k-1}\\ \hat{\boldsymbol{\theta}}_{1,k-1}\end{bmatrix}+\mu_k(t)\hat{\boldsymbol{\Gamma}}_{1,k}^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_1(L)-\hat{\boldsymbol{\Gamma}}_{1,k}(L)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{m,k-1}\\ \hat{\boldsymbol{\theta}}_{1,k-1}\end{bmatrix}\right\},\quad(70)$$
$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i-1,k}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k(t)\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_i(L)-\hat{\boldsymbol{\Gamma}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i-1,k}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},$$
$$i=2,3,\cdots,m,\quad(71)$$
$$\hat{\boldsymbol{\Gamma}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},$$
$$i=1,2,\cdots,m,\quad(72)$$
$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\quad(73)$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{74}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{75}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{76}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{77}$$

$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},\tag{78}$$

$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{m,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix},\tag{79}$$

$$\mu_k(t)\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)].\tag{80}$$

PC-GI 算法(70)—(80)输出的参数估计为 $\hat{\boldsymbol{\alpha}}_k=\hat{\boldsymbol{\alpha}}_{m,k}$，$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]$，其计算步骤如下：

① 确定数据长度 L，收集输入输出数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,\cdots,L\}$，给定参数估计精度 ε. 用式(73)构造 $\boldsymbol{Y}_i(L)$，用式(75)构造 $\boldsymbol{\varphi}(t)$.

② 置初值：令 $k=1$，$\hat{\boldsymbol{\alpha}}_{m,0}=\mathbf{1}_m/p_0$，$\hat{\boldsymbol{\theta}}_{i,0}=\mathbf{1}_{nr}/p_0(i=1,2,\cdots,m)$，$\hat{\boldsymbol{v}}_0(t)=\mathbf{1}_m/p_0$.

③ 用式(76)构造 $\hat{\boldsymbol{\phi}}_k(t)$，从式(77)中读出 $\hat{\boldsymbol{\phi}}_{i,k}(t)$，以构成式(74)中的 $\hat{\boldsymbol{\psi}}_{i,k}(t)$ 和式(72)的 $\hat{\boldsymbol{\Gamma}}_{i,k}(L)$.

④ 根据式(80)选择 $\mu_k(t)=2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\cdot\hat{\boldsymbol{\Gamma}}_{i,k}(L)]$，用式(70)和式(71)刷新参数估计 $\hat{\boldsymbol{\alpha}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$.

⑤ 用式(79)计算 $\hat{v}_{i,k}(t)$，并构造(78)中的 $\hat{\boldsymbol{v}}_k(t)$.

⑥ 比较 $\hat{\boldsymbol{\alpha}}_{i,k}$ 和 $\hat{\boldsymbol{\alpha}}_{i,k-1}$，$\hat{\boldsymbol{\theta}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k-1}$，如果 $\|\hat{\boldsymbol{\alpha}}_{m,k}-\hat{\boldsymbol{\alpha}}_{m,k-1}\|+\|\hat{\boldsymbol{\theta}}_{i,k}-\hat{\boldsymbol{\theta}}_{i,k-1}\|\leqslant\varepsilon$，就结束计算，获得迭代次数 k 和参数估计 $\hat{\boldsymbol{\alpha}}_{m,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$；否则，$k$ 增 1，转到步骤③.

2.5 子系统最小二乘迭代辨识算法

定义并极小化准则函数

$$J_6(\boldsymbol{\alpha},\boldsymbol{\vartheta}_i):=\left\|\boldsymbol{Y}_i(L)-\boldsymbol{\Gamma}_i(L)\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\theta}_i\end{bmatrix}\right\|,$$

得到估计参数向量 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\theta}_i$ 的最小二乘估计：

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}\\ \hat{\boldsymbol{\theta}}_i\end{bmatrix}=[\boldsymbol{\Gamma}_i^{\mathrm{T}}(L)\boldsymbol{\Gamma}_i(L)]^{-1}\boldsymbol{\Gamma}_i^{\mathrm{T}}(L)\boldsymbol{Y}_i(L).\tag{81}$$

辨识的困难在于 $\boldsymbol{\Gamma}_i(L)$ 中包含了不可测噪声项 $v_i(t-j)$，式(81)计算参数估计 $\hat{\boldsymbol{\alpha}}$ 和 $\hat{\boldsymbol{\theta}}_i$ 的算法无法实现. 采用辅助模型辨识思想，将未知变量 $\boldsymbol{\Gamma}_i(L)$ 和 $\boldsymbol{\phi}_i(t)$ 分别用其第 k 次迭代估计值 $\hat{\boldsymbol{\Gamma}}_{i,k}(L)$ 和 $\hat{\boldsymbol{\phi}}_{i,k}(t)$ 代替，将不可测噪声项 $v_i(t-j)$ 用其第 $k-1$ 次迭代估计值 $\hat{v}_{i,k-1}(t-j)$ 代替. 用 $\hat{\boldsymbol{\Gamma}}_{i,k}(L)$ 代替式(81)中 $\boldsymbol{\Gamma}_i(L)$，可得辨识参数向量 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\theta}_i$ 的子系统最小二乘迭代算法：

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_k\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)]^{-1}\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\boldsymbol{Y}_i(L),$$

$$i=1,2,\cdots,m,$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(1),\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(L)]^{\mathrm{T}},$$

$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},$$

$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_k\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}.$$

给每个子系统中的 $\hat{\boldsymbol{\alpha}}_k$ 加下标 i，便得到下列子系统最小二乘迭代算法(Subsystem Least Squares based Iterative algorithm，S-LSI 算法)：

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)]^{-1}\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\boldsymbol{Y}_i(L),$$

$$i=1,2,\cdots,m,\tag{82}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(1),\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(L)]^{\mathrm{T}},\tag{83}$$

$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\tag{84}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{85}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{86}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{87}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{88}$$

$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},\tag{89}$$

$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}.\tag{90}$$

2.6 部分耦合子系统最小二乘迭代辨识算法

在 S-LSI 算法(82)—(90)中，对参数向量 $\boldsymbol{\alpha}$ 进行了 m 次估计. 与 S-GI 算法的处理方法相同，取每个子系统的估计 $\hat{\boldsymbol{\alpha}}_{i,k}$ 的平均值作为参数向量 $\boldsymbol{\alpha}$ 的估计，即

$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m},\tag{91}$$

用 $\hat{\boldsymbol{\alpha}}_k$ 代替式(90)右边的 $\hat{\boldsymbol{\alpha}}_{i,k}$，就得到部分耦合子系统最小二乘迭代算法(Partially Coupled Subsystem Least Squares based Iterative algorithm，PC-S-LSI 算

法).

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Gamma}}_{i,k}(L)]^{-1}\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\boldsymbol{Y}_i(L),$$

$$i=1,2,\cdots,m, \tag{92}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(1),\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(L)]^{\mathrm{T}}, \tag{93}$$

$$\boldsymbol{Y}_i(L)=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}}, \tag{94}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}}, \tag{95}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{96}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)] \tag{97}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}}, \tag{98}$$

$$\hat{\boldsymbol{v}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{\alpha}}_k-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t), \tag{99}$$

$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m}, \tag{100}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]. \tag{101}$$

3 多变量 CARARMA 系统的迭代辨识方法

这里将上节的方法推广到多变量受控自回归自回归滑动平均系统(多变量 CARARMA 系统).

3.1 系统描述与辨识模型

考虑下列多变量 CARARMA 系统,即多变量方程误差自回归滑动平均系统(多变量 EEARMA 系统):

$$\boldsymbol{A}(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)\boldsymbol{u}(t)+\frac{d(z)}{c(z)}\boldsymbol{v}(t), \tag{102}$$

其中 $\boldsymbol{w}(t):=\frac{d(z)}{c(z)}\boldsymbol{v}(t)\in\mathbf{R}^m$ 为 ARMA 过程,$c(z)\in\mathbf{R}$ 为单位后移算子 z^{-1}的首 1 多项式:

$$c(z):=1+c_1z^{-1}+c_2z^{-2}+\cdots+c_{n_c}z^{-n_c},\quad c_i\in\mathbf{R},$$

假设 m,r,n_a,n_b,n_c 和 n_d 已知,且当 $t\leqslant0$ 时,$\boldsymbol{y}(t)=0,\boldsymbol{u}(t)=0,\boldsymbol{v}(t)=0$.辨识的目标是基于耦合辨识概念和数据滤波技术,利用系统的观测数据 $\{\boldsymbol{y}(t),\boldsymbol{u}(t):t=1,2,3,\cdots\}$ 提出新的算法,估计系统模型参数矩阵 $\boldsymbol{A}_i,\boldsymbol{B}_i$,以及噪声模型的参数 c_i 和参数 d_i.

定义系统参数矩阵 $\boldsymbol{\theta}$,参数向量 $\boldsymbol{\alpha}$,信息向量 $\boldsymbol{\varphi}(t)$ 和信息矩阵 $\boldsymbol{\phi}(t)$ 如下:

$$\boldsymbol{\theta}^{\mathrm{T}}:=[\boldsymbol{A}_1,\boldsymbol{A}_2,\cdots,\boldsymbol{A}_{n_a},\boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_{n_b}]\in\mathbf{R}^{m\times n},\\ n:=mn_a+rn_b,$$

$$\boldsymbol{\alpha}:=[c_1,c_2,\cdots,c_{n_c},d_1,d_2,\cdots,d_{n_d}]^{\mathrm{T}}\in\mathbf{R}^{n_c+n_d},$$

$$\boldsymbol{\varphi}(t):=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}\in\mathbf{R}^n,$$

$$\boldsymbol{\phi}(t):=[-\boldsymbol{w}(t-1),-\boldsymbol{w}(t-2),\cdots,-\boldsymbol{w}(t-n_c),\\ \boldsymbol{v}(t-1),\boldsymbol{v}(t-2),\cdots,\boldsymbol{v}(t-n_d)]\in\mathbf{R}^{m\times(n_c+n_d)}.$$

于是,有

$$\boldsymbol{w}(t)=[1-c(z)]\boldsymbol{w}(t)+d(z)\boldsymbol{v}(t)=\\ \boldsymbol{\phi}(t)\boldsymbol{\alpha}+\boldsymbol{v}(t), \tag{103}$$

$$\boldsymbol{w}(t)=\boldsymbol{A}(z)\boldsymbol{y}(t)-\boldsymbol{B}(z)\boldsymbol{u}(t)=\\ \boldsymbol{y}(t)-\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t), \tag{104}$$

$$\boldsymbol{y}(t)=[\boldsymbol{I}-\boldsymbol{A}(z)]\boldsymbol{y}(t)+\boldsymbol{B}(z)\boldsymbol{u}(t)+\boldsymbol{w}(t)=\\ \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t)+\boldsymbol{\phi}(t)\boldsymbol{\alpha}+\boldsymbol{v}(t)=\\ \boldsymbol{\phi}(t)\boldsymbol{\alpha}+\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t)+\boldsymbol{v}(t). \tag{105}$$

式(105)就是多变量 CARARMA 系统(102)的递阶辨识模型.

3.2 子系统梯度迭代辨识算法

令 $\boldsymbol{\phi}_i^{\mathrm{T}}(t)$ 为信息矩阵 $\boldsymbol{\phi}(t)$ 的第 i 行,$\boldsymbol{\theta}_i$ 为参数矩阵 $\boldsymbol{\theta}$ 的第 i 列,即

$$\boldsymbol{\phi}(t):=[\boldsymbol{\phi}_1(t),\boldsymbol{\phi}_2(t),\cdots,\boldsymbol{\phi}_m(t)]^{\mathrm{T}}\in\mathbf{R}^{m\times n},$$

$$\boldsymbol{\theta}:=[\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_m]\in\mathbf{R}^{(nr)\times m}.$$

定义子系统信息向量为 $\boldsymbol{\psi}_i(t):=\begin{bmatrix}\boldsymbol{\phi}_i(t)\\ \boldsymbol{\varphi}(t)\end{bmatrix}\in\mathbf{R}^{n+nr}$.参考 CARMA 系统的子辨识模型的推导,可将式(105)写为下列子系统辨识模型:

$$y_i(t)=\boldsymbol{\psi}_i^{\mathrm{T}}(t)\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\theta}_i\end{bmatrix}+v_i(t),\quad i=1,2,\cdots,m. \tag{106}$$

定义子系统堆积输出向量 $\boldsymbol{Y}_i$,堆积信息矩阵 $\boldsymbol{\Gamma}_i$,堆积白噪声向量 $\boldsymbol{V}_i$ 如下:

$$\boldsymbol{Y}_i:=\begin{bmatrix}y_i(1)\\ y_i(2)\\ \vdots\\ y_i(L)\end{bmatrix}\in\mathbf{R}^L,\quad \boldsymbol{\Gamma}_i:=\begin{bmatrix}\boldsymbol{\psi}_i^{\mathrm{T}}(1)\\ \boldsymbol{\psi}_i^{\mathrm{T}}(2)\\ \vdots\\ \boldsymbol{\psi}_i^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times(n+nr)},$$

$$\boldsymbol{V}_i:=\begin{bmatrix}v_i(1)\\ v_i(2)\\ \vdots\\ v_i(L)\end{bmatrix}\in\mathbf{R}^L.$$

由式(106)可得

$$\boldsymbol{Y}_i=\boldsymbol{\Gamma}_i\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\theta}_i\end{bmatrix}+\boldsymbol{V}_i,\quad i=1,2,\cdots,m. \tag{107}$$

定义准则函数

$$J_7(\boldsymbol{\alpha},\boldsymbol{\theta}_i):=\left\|\boldsymbol{Y}_i-\boldsymbol{\Gamma}_i\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\theta}_i\end{bmatrix}\right\|^2.$$

参照 CARMA 系统子系统梯度迭代算法(49)—(58)的推导过程,使用负梯度搜索,极小化准则函数

$J_7(\boldsymbol{\alpha},\boldsymbol{\theta}_i)$,可得估计 CARARMA 系统参数向量 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\theta}_i$ 的子系统梯度迭代算法(Subsystem Gradient based Iterative algorithm,S-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k(t)\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\left\{\boldsymbol{Y}_i-\hat{\boldsymbol{\Gamma}}_{i,k}\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},\tag{108}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}:=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\tag{109}$$

$$\boldsymbol{Y}_i=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\tag{110}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{111}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{112}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c),\ \hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{113}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{114}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{115}$$

$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},\tag{116}$$

$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix},\tag{117}$$

$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\hat{\boldsymbol{\Gamma}}_{i,k}],\tag{118}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\tag{119}$$

3.3 部分耦合子系统梯度迭代辨识算法

子系统梯度迭代辨识算法产生冗余估计 $\boldsymbol{\alpha}_{i,k}$,取它们的平均值

$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m}$$

作为 $\boldsymbol{\alpha}$ 的估计.将式(108)右边的 $\hat{\boldsymbol{\alpha}}_{i,k-1}$ 用平均值 $\hat{\boldsymbol{\alpha}}_{k-1}$ 代替,将式(117)右边的 $\hat{\boldsymbol{\alpha}}_{i,k}$ 用平均值 $\hat{\boldsymbol{\alpha}}_k$ 代替,就得到了部分耦合子系统梯度迭代算法(Partially Coupled Subsystem Gradient based Iterative algorithm,PC-S-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\left\{\boldsymbol{Y}_i-\hat{\boldsymbol{\Gamma}}_{i,k}\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},\tag{120}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}:=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\tag{121}$$

$$\boldsymbol{Y}_i=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\tag{122}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{123}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{124}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c),\ \hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{125}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{126}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{127}$$

$$\hat{\boldsymbol{v}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{\alpha}}_k-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{128}$$

$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\hat{\boldsymbol{\Gamma}}_{i,k}],\tag{129}$$

$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m},\tag{130}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\tag{131}$$

3.4 部分耦合梯度迭代辨识算法

借鉴算法(70)—(80)的推导方法,基于部分耦合子系统梯度迭代算法(108)—(119),利用耦合辨识概念,用 $\hat{\boldsymbol{\alpha}}_{m,k-1}$ 代替式(108)中 $i=1$ 时的 $\hat{\boldsymbol{\alpha}}_{1,k-1}$,用 $\hat{\boldsymbol{\alpha}}_{i-1,k}$ 代替式(108)右边的 $\hat{\boldsymbol{\alpha}}_{i,k-1}$,则得到 CARARMA 系统的部分耦合梯度迭代算法(Partially Coupled Gradient based Iterative algorithm,PC-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{1,k}\\ \hat{\boldsymbol{\theta}}_{1,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{m,k-1}\\ \hat{\boldsymbol{\theta}}_{1,k-1}\end{bmatrix}+\mu_k\hat{\boldsymbol{\Gamma}}_{1,k}^{\mathrm{T}}\left\{\boldsymbol{Y}_1-\hat{\boldsymbol{\Gamma}}_{1,k}\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{m,k-1}\\ \hat{\boldsymbol{\theta}}_{1,k-1}\end{bmatrix}\right\},\tag{132}$$

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i-1,k}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\left\{\boldsymbol{Y}_i-\hat{\boldsymbol{\Gamma}}_{i,k}\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i-1,k}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},\ i=2,3,\cdots,m,\tag{133}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\ i=1,2,\cdots,m,\tag{134}$$

$$\boldsymbol{Y}_i=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\tag{135}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{136}$$

$$\boldsymbol{\varphi}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{137}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c),\ \hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{138}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{139}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{140}$$

$$\hat{\boldsymbol{v}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{\alpha}}_{m,k}-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{141}$$

$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\hat{\boldsymbol{\Gamma}}_{i,k}],\tag{142}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\tag{143}$$

该算法输出的参数估计为 $\hat{\boldsymbol{\alpha}}_k=\hat{\boldsymbol{\alpha}}_{m,k}$,$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]$,其计算步骤如下:

① 确定数据长度 L,收集输入输出数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,\cdots,L\}$,给定参数估计精度 ε.用式(135)构造 $\boldsymbol{Y}_i(L)$,用式(137)构造 $\boldsymbol{\varphi}(t)$.

② 置初值:令 $k=1$,$\hat{\boldsymbol{\alpha}}_{m,0}=\mathbf{1}_m/p_0$,$\hat{\boldsymbol{\theta}}_{i,0}=\mathbf{1}_{nr}/p_0(i=1,2,\cdots,m)$,$\hat{\boldsymbol{w}}_0(t)=\mathbf{1}_m/p_0$,$\hat{\boldsymbol{v}}_0(t)=\mathbf{1}_m/p_0$.

③ 用式(138)构造 $\hat{\boldsymbol{\phi}}_k(t)$,从式(139)中读出 $\hat{\boldsymbol{\phi}}_{i,k}(t)$,以构成式(136)中的 $\hat{\boldsymbol{\psi}}_{i,k}(t)$ 和式(134)的

$\hat{\boldsymbol{\Gamma}}_{i,k}(L)$.

④ 根据式(142)选择 $\mu_k(t)=2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}(L)\cdot\hat{\boldsymbol{\Gamma}}_{i,k}(L)]$，用式(132)和式(133)刷新参数估计 $\hat{\boldsymbol{\alpha}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$.

⑤ 用式(140)计算 $\hat{\boldsymbol{w}}_k(t)$，用式(141)计算 $\hat{\boldsymbol{v}}_k(t)$.

⑥ 比较 $\hat{\boldsymbol{\alpha}}_{i,k}$ 和 $\hat{\boldsymbol{\alpha}}_{i,k-1}$，$\hat{\boldsymbol{\theta}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k-1}$，如果 $\|\hat{\boldsymbol{\alpha}}_{m,k}-\hat{\boldsymbol{\alpha}}_{m,k-1}\|\leqslant\varepsilon$ 和 $\|\hat{\boldsymbol{\theta}}_{i,k}-\hat{\boldsymbol{\theta}}_{i,k-1}\|\leqslant\varepsilon$，就结束计算，获得迭代次数 k 和参数估计 $\hat{\boldsymbol{\alpha}}_{m,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$；否则，$k$ 增 1，转到步骤③.

3.5 子系统最小二乘迭代辨识算法

类似地，有 CARARMA 系统的子系统最小二乘迭代算法(Subsystem Least Squares based Iterative algorithm, S-LSI 算法)：

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\hat{\boldsymbol{\Gamma}}_{i,k}]^{-1}\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\boldsymbol{Y}_i,\quad i=1,2,\cdots,m,\tag{144}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}:=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\tag{145}$$

$$\boldsymbol{Y}_i=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\tag{146}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{147}$$

$$\begin{aligned}\boldsymbol{\varphi}(t)=[&-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\\ &\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\end{aligned}\tag{148}$$

$$\begin{aligned}\hat{\boldsymbol{\phi}}_k(t)=[&-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c),\\ &\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\end{aligned}\tag{149}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{150}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{151}$$

$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}},\tag{152}$$

$$\hat{v}_{i,k}(t)=y_i(t)-\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(t)\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix},\tag{153}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\tag{154}$$

3.6 部分耦合子系统最小二乘迭代辨识算法

类似地，有 CARARMA 系统的部分耦合子系统最小二乘迭代算法(Partially Coupled Subsystem Least Squares based Iterative algorithm, PC-S-LSI 算法)：

$$\begin{bmatrix}\hat{\boldsymbol{\alpha}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\hat{\boldsymbol{\Gamma}}_{i,k}]^{-1}\hat{\boldsymbol{\Gamma}}_{i,k}^{\mathrm{T}}\boldsymbol{Y}_i,\quad i=1,2,\cdots,m,\tag{155}$$

$$\hat{\boldsymbol{\Gamma}}_{i,k}:=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\tag{156}$$

$$\boldsymbol{Y}_i=[y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}},\tag{157}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\boldsymbol{\varphi}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{158}$$

$$\begin{aligned}\boldsymbol{\varphi}(t)=[&-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\\ &\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\end{aligned}\tag{159}$$

$$\begin{aligned}\hat{\boldsymbol{\phi}}_k(t)=[&-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c),\\ &\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\end{aligned}\tag{160}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{161}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{162}$$

$$\hat{\boldsymbol{v}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{\alpha}}_k-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(t),\tag{163}$$

$$\hat{\boldsymbol{\alpha}}_k=\frac{\hat{\boldsymbol{\alpha}}_{1,k}+\hat{\boldsymbol{\alpha}}_{2,k}+\cdots+\hat{\boldsymbol{\alpha}}_{m,k}}{m},\tag{164}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\tag{165}$$

PC-S-LSI 算法(155)—(165)的计算步骤如下：

① 确定数据长度 L，收集输入输出数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=0,1,\cdots,L\}$，给定参数估计精度 ε.

② 置初值：令 $k=1$，$\hat{\boldsymbol{w}}_0(t)=$ 随机向量，$\hat{\boldsymbol{v}}_0(t)=$ 随机向量.用式(157)构造 $\boldsymbol{Y}_i(t)$，用式(159)构造 $\boldsymbol{\varphi}(t)$.

③ 用式(160)构造 $\hat{\boldsymbol{\phi}}_k(t)$，并从式(161)中读出 $\hat{\boldsymbol{\phi}}_{i,k}(t)$，以构成式(158)中的 $\hat{\boldsymbol{\psi}}_{i,k}(t)$ 和式(156)的中 $\hat{\boldsymbol{\Gamma}}_{i,k}$.

④ 用式(155)刷新参数估计 $\hat{\boldsymbol{\alpha}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$ 和用式(164)计算第 k 次的平均迭代估计值 $\hat{\boldsymbol{\alpha}}_k$，用式(165)构成参数估计 $\hat{\boldsymbol{\theta}}_k$.

⑤ 用式(162)计算 $\hat{\boldsymbol{w}}_k(t)$，用式(163)计算 $\hat{\boldsymbol{v}}_k(t)$.

⑥ 比较 $\hat{\boldsymbol{\alpha}}_k$ 和 $\hat{\boldsymbol{\alpha}}_{k-1}$，及 $\hat{\boldsymbol{\theta}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k-1}$，如果 $\|\hat{\boldsymbol{\alpha}}_k-\hat{\boldsymbol{\alpha}}_{k-1}\|\leqslant\varepsilon$，$\|\hat{\boldsymbol{\theta}}_{i,k}-\hat{\boldsymbol{\theta}}_{i,k-1}\|\leqslant\varepsilon$，就结束计算，获得迭代次数 k 和参数估计 $\hat{\boldsymbol{\alpha}}_k$ 和 $\hat{\boldsymbol{\theta}}_k$；否则 k 增 1，转到步骤③.

4 基于滤波的多变量 CARARMA 系统的迭代辨识方法

本节利用(半)滤波技术，结合耦合辨识概念，研究多变量 CARARMA 系统的迭代辨识方法.

4.1 基于滤波的子系统辨识模型

考虑式(102)描述的多变量 CARARMA 系统，重写如下：

$$\boldsymbol{A}(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)\boldsymbol{u}(t)+\frac{d(z)}{c(z)}\boldsymbol{v}(t),\tag{166}$$

其中各变量定义同上，$\boldsymbol{w}(t):=\dfrac{d(z)}{c(z)}\boldsymbol{v}(t)$.

定义滤波输入向量 $\boldsymbol{u}_{\mathrm{f}}(t)$ 和滤波输出向量 $\boldsymbol{y}_{\mathrm{f}}(t)$ 为

$$\boldsymbol{u}_{\mathrm{f}}(t):=c(z)\boldsymbol{u}(t)\in\mathbf{R}^r,$$

$$\boldsymbol{y}_{\mathrm{f}}(t):=c(z)\boldsymbol{y}(t)\in\mathbf{R}^{m}.$$

式(166)两边同时左乘 $c(z)$ 得到

$$\boldsymbol{A}(z)c(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)c(z)\boldsymbol{u}(t)+d(z)\boldsymbol{v}(t),$$

或

$$\boldsymbol{A}(z)\boldsymbol{y}_{\mathrm{f}}(t)=\boldsymbol{B}(z)\boldsymbol{u}_{\mathrm{f}}(t)+d(z)\boldsymbol{v}(t). \tag{167}$$

定义系统参数矩阵 $\boldsymbol{\theta}$,噪声参数向量 $\boldsymbol{d}$,滤波信息向量 $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ 和信息矩阵 $\boldsymbol{\phi}(t)$ 如下:

$$\boldsymbol{\theta}^{\mathrm{T}}:=[\boldsymbol{A}_1,\boldsymbol{A}_2,\cdots,\boldsymbol{A}_{n_a},\boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_{n_b}]\in\mathbf{R}^{m\times n},$$

$$n:=mn_a+rn_b,$$

$$\boldsymbol{d}:=[d_1,d_2,\cdots,d_{n_d}]^{\mathrm{T}}\in\mathbf{R}^{n_d},$$

$$\boldsymbol{\varphi}_{\mathrm{f}}(t):=[-\boldsymbol{y}_{\mathrm{f}}^{\mathrm{T}}(t-1),-\boldsymbol{y}_{\mathrm{f}}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}_{\mathrm{f}}^{\mathrm{T}}(t-n_a), \boldsymbol{u}_{\mathrm{f}}^{\mathrm{T}}(t-1),\boldsymbol{u}_{\mathrm{f}}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}_{\mathrm{f}}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}\in\mathbf{R}^{n},$$

$$\boldsymbol{\phi}(t):=[\boldsymbol{v}(t-1),\boldsymbol{v}(t-2),\cdots,\boldsymbol{v}(t-n_d)]\in\mathbf{R}^{m\times n_d}.$$

则式(167)可以改写成

$$\boldsymbol{y}_{\mathrm{f}}(t)=[\boldsymbol{I}-\boldsymbol{A}(z)]\boldsymbol{y}_{\mathrm{f}}(t)+\boldsymbol{B}(z)\boldsymbol{u}_{\mathrm{f}}(t)+d(z)\boldsymbol{v}(t)= \boldsymbol{\phi}(t)\boldsymbol{d}+\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{f}}(t)+\boldsymbol{v}(t). \tag{168}$$

令 $y_{\mathrm{f}i}(t)$ 为滤波输出向量 $\boldsymbol{y}_{\mathrm{f}}(t)$ 的第 i 元,$\boldsymbol{\phi}_i^{\mathrm{T}}(t)$ 为信息矩阵 $\boldsymbol{\phi}(t)$ 的第 i 行,$\boldsymbol{\theta}_i$ 为参数矩阵 $\boldsymbol{\theta}$ 的第 i 列,即

$$\boldsymbol{y}_{\mathrm{f}}(t):=[y_{\mathrm{f}1}(t),y_{\mathrm{f}2}(t),\cdots,y_{\mathrm{f}m}(t)]^{\mathrm{T}}\in\mathbf{R}^{m},$$

$$\boldsymbol{\phi}(t):=[\boldsymbol{\phi}_1(t),\boldsymbol{\phi}_2(t),\cdots,\boldsymbol{\phi}_m(t)]^{\mathrm{T}}\in\mathbf{R}^{m\times n_d},$$

$$\boldsymbol{\theta}:=[\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_m]\in\mathbf{R}^{n\times m}.$$

将递阶辨识模型(168)进一步写为

$$\begin{bmatrix} y_{\mathrm{f}1}(t)\\ y_{\mathrm{f}2}(t)\\ \vdots\\ y_{\mathrm{f}m}(t)\end{bmatrix}=\begin{bmatrix}\boldsymbol{\phi}_1^{\mathrm{T}}(t)\\ \boldsymbol{\phi}_2^{\mathrm{T}}(t)\\ \vdots\\ \boldsymbol{\phi}_m^{\mathrm{T}}(t)\end{bmatrix}\boldsymbol{d}+\begin{bmatrix}\boldsymbol{\theta}_1^{\mathrm{T}}\\ \boldsymbol{\theta}_2^{\mathrm{T}}\\ \vdots\\ \boldsymbol{\theta}_m^{\mathrm{T}}\end{bmatrix}\boldsymbol{\varphi}_{\mathrm{f}}(t)+\begin{bmatrix}v_1(t)\\ v_2(t)\\ \vdots\\ v_m(t)\end{bmatrix},$$

则式(168)可以分解为 m 个子辨识模型:

$$y_{\mathrm{f}i}(t)=\boldsymbol{\phi}_i^{\mathrm{T}}(t)\boldsymbol{d}+\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{f}}(t)+v_i(t)= \boldsymbol{\phi}_i^{\mathrm{T}}(t)\boldsymbol{d}+\boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\boldsymbol{\theta}_i+v_i(t)= [\boldsymbol{\phi}_i^{\mathrm{T}}(t),\boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)]\begin{bmatrix}\boldsymbol{d}\\ \boldsymbol{\theta}_i\end{bmatrix}+v_i(t),\quad i=1,2,\cdots,m.$$

定义子系统信息向量为 $\boldsymbol{\psi}_i(t):=\begin{bmatrix}\boldsymbol{\phi}_i(t)\\ \boldsymbol{\varphi}_{\mathrm{f}}(t)\end{bmatrix}\in\mathbf{R}^{n+n_d}$,则有

$$y_{\mathrm{f}i}(t)=\boldsymbol{\psi}_i^{\mathrm{T}}(t)\begin{bmatrix}\boldsymbol{d}\\ \boldsymbol{\theta}_i\end{bmatrix}+v_i(t). \tag{169}$$

由 $\boldsymbol{w}(t)$ 的定义可知

$$c(z)\boldsymbol{w}(t)=d(z)\boldsymbol{v}(t). \tag{170}$$

定义噪声参数向量 $\boldsymbol{c}$ 和噪声信息矩阵 $\boldsymbol{\Omega}(t)$ 为

$$\boldsymbol{c}:=[c_1,c_2,\cdots,c_{n_c}]^{\mathrm{T}}\in\mathbf{R}^{n_c},$$

$$\boldsymbol{\Omega}(t):=[-\boldsymbol{w}(t-1),-\boldsymbol{w}(t-2),\cdots,-\boldsymbol{w}(t-n_c)]\in\mathbf{R}^{m\times n_c},$$

则由式(170)有

$$\boldsymbol{w}(t)=\boldsymbol{\Omega}(t)\boldsymbol{c}+\boldsymbol{\phi}(t)\boldsymbol{d}+\boldsymbol{v}(t).$$

定义中间变量 $\boldsymbol{w}_{\mathrm{n}}(t):=\boldsymbol{w}(t)-\boldsymbol{\phi}(t)\boldsymbol{d}\in\mathbf{R}^{m}$.上式可写为

$$\boldsymbol{w}_{\mathrm{n}}(t)=\boldsymbol{\Omega}(t)\boldsymbol{c}+\boldsymbol{v}(t). \tag{171}$$

式(169)和式(171)即为多变量 CARARMA 系统(167)的基于滤波的递阶子系统辨识模型和噪声模型.

4.2 基于滤波的子系统梯度迭代辨识算法

考虑数据长度为 L 的一组数据,定义子系统堆积滤波输出向量 $\boldsymbol{Y}_{\mathrm{f}i}(L)$,堆积信息矩阵 $\boldsymbol{\Psi}_i(L)$,堆积白噪声向量 $\boldsymbol{V}_i(L)$,堆积噪声输出向量 $\boldsymbol{W}_{\mathrm{n}}(L)$,堆积噪声信息矩阵 $\boldsymbol{\Upsilon}(L)$ 和堆积白噪声向量 $\boldsymbol{V}(L)$ 如下:

$$\boldsymbol{Y}_{\mathrm{f}i}(L):=\begin{bmatrix}y_{\mathrm{f}i}(1)\\ y_{\mathrm{f}i}(2)\\ \vdots\\ y_{\mathrm{f}i}(L)\end{bmatrix}\in\mathbf{R}^{L},\quad \boldsymbol{\Psi}_i(L):=\begin{bmatrix}\boldsymbol{\psi}_i^{\mathrm{T}}(1)\\ \boldsymbol{\psi}_i^{\mathrm{T}}(2)\\ \vdots\\ \boldsymbol{\psi}_i^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times(n+n_d)},$$

$$\boldsymbol{V}_i(L):=\begin{bmatrix}v_i(1)\\ v_i(2)\\ \vdots\\ v_i(L)\end{bmatrix}\in\mathbf{R}^{L},\quad \boldsymbol{W}_{\mathrm{n}}(L):=\begin{bmatrix}\boldsymbol{w}_{\mathrm{n}}(1)\\ \boldsymbol{w}_{\mathrm{n}}(2)\\ \vdots\\ \boldsymbol{w}_{\mathrm{n}}(L)\end{bmatrix}\in\mathbf{R}^{mL},$$

$$\boldsymbol{\Upsilon}(L):=\begin{bmatrix}\boldsymbol{\Omega}(1)\\ \boldsymbol{\Omega}(2)\\ \vdots\\ \boldsymbol{\Omega}(L)\end{bmatrix}\in\mathbf{R}^{(mL)\times n_c},\quad \boldsymbol{V}(L):=\begin{bmatrix}\boldsymbol{v}(1)\\ \boldsymbol{v}(2)\\ \vdots\\ \boldsymbol{v}(L)\end{bmatrix}\in\mathbf{R}^{mL},$$

由式(169)和式(171)可得

$$\boldsymbol{Y}_{\mathrm{f}i}(L)=\boldsymbol{\Psi}_i(L)\begin{bmatrix}\boldsymbol{d}\\ \boldsymbol{\theta}_i\end{bmatrix}+\boldsymbol{V}_i(L),\quad i=1,2,\cdots,m,$$

$$\boldsymbol{W}_{\mathrm{n}}(L)=\boldsymbol{\Upsilon}(L)\boldsymbol{c}+\boldsymbol{V}(L).$$

令 $\hat{\boldsymbol{d}}_k$,$\hat{\boldsymbol{\theta}}_{i,k}$ 和 $\hat{\boldsymbol{c}}_k$ 分别为 $\boldsymbol{d}$,$\boldsymbol{\theta}_i$ 和 $\boldsymbol{c}$ 在第 k 次迭代的参数估计,定义并极小化准则函数

$$J_8(\boldsymbol{d},\boldsymbol{\theta}_i):=\left\|\boldsymbol{Y}_{\mathrm{f}i}(L)-\boldsymbol{\Psi}_i(L)\begin{bmatrix}\boldsymbol{d}\\ \boldsymbol{\theta}_i\end{bmatrix}\right\|^2,$$

$$J_9(\boldsymbol{c}):=\|\boldsymbol{W}_{\mathrm{n}}(L)-\boldsymbol{\Upsilon}(L)\boldsymbol{c}\|^2.$$

得到估计参数向量 $\boldsymbol{d}$,$\boldsymbol{\theta}_i$ 和 $\boldsymbol{c}$ 基于滤波的梯度迭代算法:

$$\begin{bmatrix}\hat{\boldsymbol{d}}_k\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{d}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}-\frac{\mu_k}{2}\mathrm{grad}[J_8(\hat{\boldsymbol{d}}_{k-1},\hat{\boldsymbol{\theta}}_{i,k-1})]= \begin{bmatrix}\hat{\boldsymbol{d}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_k\boldsymbol{\Psi}_i^{\mathrm{T}}(L)\left\{\boldsymbol{Y}_{\mathrm{f}i}(L)-\boldsymbol{\Psi}_i(L)\begin{bmatrix}\hat{\boldsymbol{d}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\}, \tag{172}$$

$$\hat{\boldsymbol{c}}_k=\hat{\boldsymbol{c}}_{k-1}-\frac{\mu_k}{2}\mathrm{grad}[J_9(\hat{\boldsymbol{c}}_{k-1})]=$$
$$\hat{\boldsymbol{c}}_{k-1}+\mu_k\boldsymbol{Y}^{\mathrm{T}}(L)[\boldsymbol{W}_{\mathrm{n}}(L)-\boldsymbol{Y}(L)\hat{\boldsymbol{c}}_{k-1}],\qquad(173)$$

其中 μ_k 为迭代步长.式(172)右边 $\boldsymbol{\Psi}_i(L)$ 中包含了不可测噪声项 $v_i(t-j)$,式(173)$\boldsymbol{Y}(L)$ 中包含了不可测噪声项 $\boldsymbol{w}(t-j)$.多项式 $c(z)$ 是未知的,故 $\boldsymbol{y}_{\mathrm{f}}(t)$ 和 $\boldsymbol{u}_{\mathrm{f}}(t)$ 也是未知的,即 $\boldsymbol{Y}_{\mathrm{f}i}(L)$ 是未知的.输出矩阵 $\boldsymbol{W}_{\mathrm{n}}(L)$ 中包含了不可测项 $\boldsymbol{w}(t-j)$,不可能利用式(172)和(173)估计 $\hat{\boldsymbol{d}}_k$,$\hat{\boldsymbol{\theta}}_{i,k}$ 和 $\hat{\boldsymbol{c}}_k$.解决方案是采用辅助模型思想,将未知变量 $\boldsymbol{\Psi}_i(L)$ 和 $\boldsymbol{Y}(L)$ 分别用其第 k 次迭代估计值 $\hat{\boldsymbol{\Psi}}_{i,k}(L)$ 和 $\hat{\boldsymbol{Y}}_k(L)$ 代替,$\boldsymbol{W}_{\mathrm{n}}(L)$ 用其估计 $\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)$ 代替.未知的滤波向量 $\boldsymbol{y}_{\mathrm{f}}(t)$ 和 $\boldsymbol{u}_{\mathrm{f}}(t)$ 用其迭代估计值 $\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)$ 和 $\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)$ 代替.由式(166)可得

$$\boldsymbol{A}(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)\boldsymbol{u}(t)+\boldsymbol{w}(t).$$

定义信息向量 $\boldsymbol{\varphi}_{\mathrm{s}}(t)$ 为

$$\boldsymbol{\varphi}_{\mathrm{s}}(t):=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}\in\mathbf{R}^n,$$

则有

$$\boldsymbol{y}(t)=\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t)+\boldsymbol{w}(t).\qquad(174)$$

式(174)右边的 $\boldsymbol{\theta}$ 用其第 k 步的迭代估计 $\hat{\boldsymbol{\theta}}_k$ 代替,可得到 $\boldsymbol{w}(t)$ 的估计值为

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t).$$

用 $\hat{\boldsymbol{w}}_{k-1}(t-j)$ 构造信息矩阵 $\boldsymbol{\Omega}(t)$ 的估计:

$$\hat{\boldsymbol{\Omega}}_k(t):=[-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c)]\in\mathbf{R}^{m\times n_c}.$$

$\boldsymbol{w}_{\mathrm{n}}(t)$ 的估计可用下式计算:

$$\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_{k-1}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_{k-1}.$$

令 $\hat{\boldsymbol{\Psi}}_{i,k}(L)$ 为 $\boldsymbol{\Psi}_i(L)$ 的估计.定义

$$\hat{\boldsymbol{\Psi}}_{i,k}(L):=\begin{bmatrix}\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(1)\\\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(2)\\\vdots\\\hat{\boldsymbol{\psi}}_{i,k}^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times(n+n_d)},$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t):=\begin{bmatrix}\hat{\boldsymbol{\phi}}_{i,k}(t)\\\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)\end{bmatrix}\in\mathbf{R}^{n+n_d},$$

$$\hat{\boldsymbol{\phi}}_k(t):=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}}\in\mathbf{R}^{m\times n_d}.$$

由式(168),$\boldsymbol{v}(t)$ 的第 k 次迭代估计值 $\hat{\boldsymbol{v}}_k(t)$ 可用下式计算:

$$\hat{\boldsymbol{v}}_k(t)=\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_k-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t).\qquad(175)$$

令

$$\hat{\boldsymbol{c}}_k:=[\hat{c}_{1,k},\hat{c}_{2,k},\cdots,\hat{c}_{n_c,k}]^{\mathrm{T}}\in\mathbf{R}^{n_c},$$

$$\hat{C}_k(z):=1+\hat{c}_{1,k}z^{-1}+\hat{c}_{2,k}z^{-2}+\cdots+\hat{c}_{n_c,k}z^{-n_c}\in\mathbf{R}.$$

用 $\hat{C}_k(z)$ 对 $\boldsymbol{u}(t)$ 和 $\boldsymbol{y}(t)$ 滤波,得到相应的估计:

$$\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)=\hat{C}_k(z)\boldsymbol{u}(t),$$
$$\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)=\hat{C}_k(z)\boldsymbol{y}(t).$$

它们也可由式(185)和(187)递归计算得到.令 $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ 在第 k 步的迭代估计为

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)=[-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-1),-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-n_a),\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-1),\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}.$$

定义

$$\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L):=\begin{bmatrix}\hat{y}_{\mathrm{f}i,k}(1)\\\hat{y}_{\mathrm{f}i,k}(2)\\\vdots\\\hat{y}_{\mathrm{f}i,k}(L)\end{bmatrix}\in\mathbf{R}^L,\quad\hat{\boldsymbol{W}}_{\mathrm{n},k}(L):=\begin{bmatrix}\hat{\boldsymbol{w}}_{\mathrm{n},k}(1)\\\hat{\boldsymbol{w}}_{\mathrm{n},k}(2)\\\vdots\\\hat{\boldsymbol{w}}_{\mathrm{n},k}(L)\end{bmatrix}\in\mathbf{R}^{mL},$$

$$\hat{Y}_k(L):=\begin{bmatrix}\hat{\boldsymbol{\Omega}}_k(1)\\\hat{\boldsymbol{\Omega}}_k(2)\\\vdots\\\hat{\boldsymbol{\Omega}}_k(L)\end{bmatrix}\in\mathbf{R}^{(mL)\times n_c},$$

$$\hat{\boldsymbol{y}}_{\mathrm{f},k}(t):=[\hat{y}_{\mathrm{f}1,k}(t),\hat{y}_{\mathrm{f}2,k}(t),\cdots,\hat{y}_{\mathrm{f}m,k}(t)]^{\mathrm{T}}\in\mathbf{R}^m,$$

$$\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_{k-1}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_{k-1},$$

$$\hat{\boldsymbol{\Omega}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),-\hat{\boldsymbol{w}}_{k-1}(t-2),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c)].$$

在式(172)中,用 $\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)$ 代替 $\boldsymbol{Y}_{\mathrm{f}i}(L)$,用 $\hat{\boldsymbol{\Psi}}_{i,k}(L)$ 代替 $\boldsymbol{\Psi}_i(L)$,在式(173)中,用 $\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)$ 代替 $\boldsymbol{W}_{\mathrm{n}}(L)$,用 $\hat{\boldsymbol{Y}}_k(L)$ 代替 $\boldsymbol{Y}(L)$,可得多变量CARARMA系统的基于滤波的子系统梯度迭代算法(Filtering based Subsystem Gradient Iterative algorithm,F-S-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{d}}_{i,k}\\\hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{d}}_{i,k-1}\\\hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_{i,k}\hat{\boldsymbol{\Psi}}_{i,k}^{\mathrm{T}}(L)\left\{\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)-\hat{\boldsymbol{\Psi}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{d}}_{i,k-1}\\\hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\},$$
$$i=1,2,\cdots,m,\qquad(176)$$

$$\hat{\boldsymbol{c}}_k=\hat{\boldsymbol{c}}_{k-1}+\mu_k\hat{\boldsymbol{Y}}_k^{\mathrm{T}}(L)[\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)-\hat{\boldsymbol{Y}}_k(L)\hat{\boldsymbol{c}}_{k-1}],\qquad(177)$$

$$\hat{\boldsymbol{\Psi}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\qquad(178)$$

$$\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)=[\hat{y}_{\mathrm{f}i,k}(1),\hat{y}_{\mathrm{f}i,k}(2),\cdots,\hat{y}_{\mathrm{f}i,k}(L)]^{\mathrm{T}},\qquad(179)$$

$$\hat{\boldsymbol{Y}}_k(L)=\begin{bmatrix}\hat{\boldsymbol{\Omega}}_k(1)\\\hat{\boldsymbol{\Omega}}_k(2)\\\vdots\\\hat{\boldsymbol{\Omega}}_k(L)\end{bmatrix},\quad\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)=\begin{bmatrix}\hat{\boldsymbol{w}}_{\mathrm{n},k}(1)\\\hat{\boldsymbol{w}}_{\mathrm{n},k}(2)\\\vdots\\\hat{\boldsymbol{w}}_{\mathrm{n},k}(L)\end{bmatrix},\qquad(180)$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}^{\mathrm{T}}(t)]^{\mathrm{T}},\qquad(181)$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\qquad(182)$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}}, \tag{183}$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)=[-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-1),-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-n_a), \boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-1),\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{184}$$

$$\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)=\boldsymbol{y}(t)+\hat{c}_{1,k}\boldsymbol{y}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{y}(t-n_c) \tag{185}$$

$$=[\hat{y}_{\mathrm{f}1,k}(t),\hat{y}_{\mathrm{f}2,k}(t),\cdots,\hat{y}_{\mathrm{f}m,k}(t)]^{\mathrm{T}}, \tag{186}$$

$$\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)=\boldsymbol{u}(t)+\hat{c}_{1,k}\boldsymbol{u}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{u}(t-n_c), \tag{187}$$

$$\hat{\boldsymbol{\Omega}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c)], \tag{188}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t), \tag{189}$$

$$\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_{k-1}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_{i,k-1}, \tag{190}$$

$$\boldsymbol{\varphi}_{\mathrm{s}}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a), \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{191}$$

$$\hat{\boldsymbol{v}}_k(t)=[\hat{v}_{1,k}(t),\hat{v}_{2,k}(t),\cdots,\hat{v}_{m,k}(t)]^{\mathrm{T}}, \tag{192}$$

$$\hat{v}_{i,k}(t)=\hat{y}_{\mathrm{f}i,k}(t)-\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t)\hat{\boldsymbol{d}}_{i,k}-\hat{\boldsymbol{\theta}}_{i,k}^{\mathrm{T}}\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t), \tag{193}$$

$$\mu_{i,k}\leqslant 2\lambda_{\max}^{-1}[\boldsymbol{\Psi}_{i,k}^{\mathrm{T}}(L)\boldsymbol{\Psi}_{i,k}(L)], \tag{194}$$

$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\boldsymbol{Y}_k^{\mathrm{T}}(L)\boldsymbol{Y}_k(L)], \tag{195}$$

$$\hat{\boldsymbol{c}}_k=[\hat{c}_{1,k},\hat{c}_{2,k},\cdots,\hat{c}_{n_c,k}]^{\mathrm{T}}, \tag{196}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]. \tag{197}$$

4.3 基于滤波的部分耦合子系统梯度迭代辨识算法

在 S-F-GI 算法(176)—(177)中,对参数向量 $\boldsymbol{d}$ 进行了 m 次估计,得到了 m 个 $\hat{\boldsymbol{d}}_{i,k}$,取其平均值作为对应参数向量的估计,即

$$\hat{\boldsymbol{d}}_k=\frac{\hat{\boldsymbol{d}}_{1,k}+\hat{\boldsymbol{d}}_{2,k}+\cdots+\hat{\boldsymbol{d}}_{m,k}}{m}\in\mathbf{R}^{n_d}.$$

用平均值 $\hat{\boldsymbol{d}}_{k-1}$ 代替式(176)和(190)右边的 $\hat{\boldsymbol{d}}_{i,k-1}$,用平均值 $\hat{\boldsymbol{d}}_k$ 代替式(193)右边的 $\hat{\boldsymbol{d}}_{i,k}$,就得到了基于滤波的部分耦合子系统梯度迭代算法(Filtering based Partially Coupled Subsystem Gradient Iterative algorithm,F-PC-S-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{d}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{d}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_{i,k}\hat{\boldsymbol{\Psi}}_{i,k}^{\mathrm{T}}(L)\left\{\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)-\hat{\boldsymbol{\Psi}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{d}}_{k-1}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\}, \tag{198}$$

$$\hat{\boldsymbol{c}}_k=\hat{\boldsymbol{c}}_{k-1}+\mu_k\hat{\boldsymbol{Y}}_k^{\mathrm{T}}(L)[\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)-\hat{\boldsymbol{Y}}_k(L)\hat{\boldsymbol{c}}_{k-1}], \tag{199}$$

$$\hat{\boldsymbol{\Psi}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}}, \tag{200}$$

$$\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)=[\hat{y}_{\mathrm{f}i,k}(1),\hat{y}_{\mathrm{f}i,k}(2),\cdots,\hat{y}_{\mathrm{f}i,k}(L)]^{\mathrm{T}}, \tag{201}$$

$$\hat{\boldsymbol{Y}}_k(L)=\begin{bmatrix}\hat{\boldsymbol{\Omega}}_k(1)\\ \hat{\boldsymbol{\Omega}}_k(2)\\ \vdots\\ \hat{\boldsymbol{\Omega}}_k(L)\end{bmatrix},\quad \hat{\boldsymbol{W}}_{\mathrm{n},k}(L)=\begin{bmatrix}\hat{\boldsymbol{w}}_{\mathrm{n},k}(1)\\ \hat{\boldsymbol{w}}_{\mathrm{n},k}(2)\\ \vdots\\ \hat{\boldsymbol{w}}_{\mathrm{n},k}(L)\end{bmatrix}, \tag{202}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}^{\mathrm{T}}(t)]^{\mathrm{T}}, \tag{203}$$

$$\hat{\boldsymbol{\phi}}_k(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)] \tag{204}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}}, \tag{205}$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)=[-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-1),-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-n_a), \boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-1),\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{206}$$

$$\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)=\boldsymbol{y}(t)+\hat{c}_{1,k}\boldsymbol{y}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{y}(t-n_c) \tag{207}$$

$$=[\hat{y}_{\mathrm{f}1,k}(t),\hat{y}_{\mathrm{f}2,k}(t),\cdots,\hat{y}_{\mathrm{f}m,k}(t)]^{\mathrm{T}}, \tag{208}$$

$$\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)=\boldsymbol{u}(t)+\hat{c}_{1,k}\boldsymbol{u}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{u}(t-n_c), \tag{209}$$

$$\hat{\boldsymbol{\Omega}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c)], \tag{210}$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t), \tag{211}$$

$$\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_{k-1}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_{k-1}, \tag{212}$$

$$\boldsymbol{\varphi}_{\mathrm{s}}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a), \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}, \tag{213}$$

$$\hat{\boldsymbol{v}}_k(t)=\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_k-\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t), \tag{214}$$

$$\mu_{i,k}\leqslant 2\lambda_{\max}^{-1}[\boldsymbol{\Psi}_{i,k}^{\mathrm{T}}(L)\boldsymbol{\Psi}_{i,k}(L)], \tag{215}$$

$$\mu_k\leqslant 2\lambda_{\max}^{-1}[\boldsymbol{Y}_k^{\mathrm{T}}(L)\boldsymbol{Y}_k(L)], \tag{216}$$

$$\hat{\boldsymbol{d}}_k=\frac{\hat{\boldsymbol{d}}_{1,k}+\hat{\boldsymbol{d}}_{2,k}+\cdots+\hat{\boldsymbol{d}}_{m,k}}{m}, \tag{217}$$

$$\hat{\boldsymbol{c}}_k=[\hat{c}_{1,k},\hat{c}_{2,k},\cdots,\hat{c}_{n_c,k}]^{\mathrm{T}}, \tag{218}$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]. \tag{219}$$

4.4 基于滤波的部分耦合梯度迭代辨识算法

类比耦合梯度迭代辨识算法,用 $\hat{\boldsymbol{d}}_{m,k-1}$ 代替式(176)中 $i=1$ 时的 $\hat{\boldsymbol{d}}_{1,k-1}$,用 $\hat{\boldsymbol{d}}_{i-1,k}$ 代替式(176)右边的 $\hat{\boldsymbol{d}}_{i,k-1}$,用 $\hat{\boldsymbol{d}}_{m,k}$ 代替式(193)右边的 $\hat{\boldsymbol{d}}_{i,k}$,则得到基于滤波的部分耦合梯度迭代算法(Filtering based Partially Coupled Gradient Iterative algorithm,F-PC-GI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{d}}_{1,k}\\ \hat{\boldsymbol{\theta}}_{1,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{d}}_{m,k-1}\\ \hat{\boldsymbol{\theta}}_{1,k-1}\end{bmatrix}+\mu_{1,k}\hat{\boldsymbol{\Psi}}_{1,k}^{\mathrm{T}}(L)\left\{\hat{\boldsymbol{Y}}_{\mathrm{f}1,k}(L)-\hat{\boldsymbol{\Psi}}_{1,k}(L)\begin{bmatrix}\hat{\boldsymbol{d}}_{m,k-1}\\ \hat{\boldsymbol{\theta}}_{1,k-1}\end{bmatrix}\right\}, \tag{220}$$

$$\begin{bmatrix}\hat{\boldsymbol{d}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=\begin{bmatrix}\hat{\boldsymbol{d}}_{i-1,k}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}+\mu_{i,k}\hat{\boldsymbol{\Psi}}_{i,k}^{\mathrm{T}}(L)\left\{\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)-\hat{\boldsymbol{\Psi}}_{i,k}(L)\begin{bmatrix}\hat{\boldsymbol{d}}_{i-1,k}\\ \hat{\boldsymbol{\theta}}_{i,k-1}\end{bmatrix}\right\}, \quad i=2,3,\cdots,m, \tag{221}$$

$$\hat{\boldsymbol{c}}_k=\hat{\boldsymbol{c}}_{k-1}+\mu_k\hat{\boldsymbol{Y}}_k^{\mathrm{T}}(L)[\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)-\hat{\boldsymbol{Y}}_k(L)\hat{\boldsymbol{c}}_{k-1}], \tag{222}$$

$$\hat{\boldsymbol{\Psi}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}}, \quad i=1,2,\cdots,m, \tag{223}$$

$$\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)=[\hat{y}_{\mathrm{f}i,k}(1),\hat{y}_{\mathrm{f}i,k}(2),\cdots,\hat{y}_{\mathrm{f}i,k}(L)]^{\mathrm{T}}, \tag{224}$$

$$\hat{Y}_k(L)=\begin{bmatrix}\hat{\boldsymbol{\Omega}}_k(1)\\ \hat{\boldsymbol{\Omega}}_k(2)\\ \vdots\\ \hat{\boldsymbol{\Omega}}_k(L)\end{bmatrix},\quad \hat{\boldsymbol{W}}_{\mathrm{n},k}(L)=\begin{bmatrix}\hat{\boldsymbol{w}}_{\mathrm{n},k}(1)\\ \hat{\boldsymbol{w}}_{\mathrm{n},k}(2)\\ \vdots\\ \hat{\boldsymbol{w}}_{\mathrm{n},k}(L)\end{bmatrix}, \tag{225}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{226}$$

$$\hat{\boldsymbol{\phi}}_{k}(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{227}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\tag{228}$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)=[-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-1),-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,-\hat{\boldsymbol{y}}_{\mathrm{f},k}^{\mathrm{T}}(t-n_a),$$

$$\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-1),\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}_{\mathrm{f},k}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{229}$$

$$\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)=\boldsymbol{y}(t)+\hat{c}_{1,k}\boldsymbol{y}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{y}(t-n_c)\tag{230}$$

$$=[\hat{y}_{\mathrm{f}1,k}(t),\hat{y}_{\mathrm{f}2,k}(t),\cdots,\hat{y}_{\mathrm{f}m,k}(t)]^{\mathrm{T}},\tag{231}$$

$$\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)=\boldsymbol{u}(t)+\hat{c}_{1,k}\boldsymbol{u}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{u}(t-n_c),\tag{232}$$

$$\hat{\boldsymbol{\Omega}}_{k}(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c)],\tag{233}$$

$$\hat{\boldsymbol{w}}_{k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_{k}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t),\tag{234}$$

$$\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}_{k-1}^{\mathrm{T}}\boldsymbol{\varphi}_{\mathrm{s}}(t)-\hat{\boldsymbol{\phi}}_{k}(t)\hat{\boldsymbol{d}}_{m,k-1},\tag{235}$$

$$\boldsymbol{\varphi}_{\mathrm{s}}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$

$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\tag{236}$$

$$\hat{\boldsymbol{v}}_{k}(t)=\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)-\hat{\boldsymbol{\phi}}_{k}(t)\hat{\boldsymbol{d}}_{m,k}-\hat{\boldsymbol{\theta}}_{k}^{\mathrm{T}}\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t),\tag{237}$$

$$\mu_{i,k}\leqslant 2\lambda_{\max}^{-1}[\boldsymbol{\Psi}_{i,k}^{\mathrm{T}}(L)\boldsymbol{\Psi}_{i,k}(L)],\tag{238}$$

$$\mu_{k}\leqslant 2\lambda_{\max}^{-1}[\boldsymbol{Y}_{k}^{\mathrm{T}}(L)\boldsymbol{Y}_{k}(L)],\tag{239}$$

$$\hat{\boldsymbol{c}}_{k}=[\hat{c}_{1,k},\hat{c}_{2,k},\cdots,\hat{c}_{n_c,k}]^{\mathrm{T}},\tag{240}$$

$$\hat{\boldsymbol{\theta}}_{k}=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\tag{241}$$

F-PC-GI 算法(220)—(241)输出的参数估计为 $\hat{\boldsymbol{d}}_k=\hat{\boldsymbol{d}}_{m,k}$, $\hat{\boldsymbol{c}}_k$, $\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]$,其计算步骤如下:

① 确定数据长度 L,收集输入输出数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,\cdots,L\}$,给定参数估计精度 ε,用式(236)构造信息向量 $\boldsymbol{\varphi}_{\mathrm{s}}(t)$.

② 置初值:令 $k=1$, $\hat{\boldsymbol{c}}_0=\mathbf{1}_{n_c}/p_0$, $\hat{\boldsymbol{d}}_{m,0}=\mathbf{1}_{n_d}/p_0$, $\hat{\boldsymbol{\theta}}_{m,0}=\boldsymbol{I}_{n\times m}/p_0$, $\hat{\boldsymbol{y}}_{\mathrm{f},0}(t)=\mathbf{1}_m/p_0$, $\hat{\boldsymbol{u}}_{\mathrm{f},0}(t)=\mathbf{1}_r/p_0$, $\hat{\boldsymbol{w}}_0(t)=\mathbf{1}_{n_c}/p_0$.

③ 用式(233)构造 $\hat{\boldsymbol{\Omega}}_k(t)$,以构成式(225)中的 $\hat{\boldsymbol{Y}}_k(L)$;用式(235)计算 $\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)$,以构成式(225)中的 $\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)$.

④ 根据式(239)选择 $\mu_k=2\lambda_{\max}^{-1}[\boldsymbol{Y}_k^{\mathrm{T}}(L)\cdot\boldsymbol{Y}_k(L)]$,用式(222)刷新参数估计向量 $\hat{\boldsymbol{c}}_k$.

⑤ 从式(240)读出 $\hat{c}_{j,k}$,用式(230)计算 $\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)$,用式(232)计算 $\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)$,用式(229)构造 $\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)$;从(231)中读取 $\hat{y}_{\mathrm{f}i,k}(t)$,以构造式(224)中的 $\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)$.

⑥ 用式(227)构造 $\hat{\boldsymbol{\phi}}_k(t)$,并从(228)中读取 $\hat{\boldsymbol{\phi}}_{i,k}(t)$,以构成式(226)中的 $\hat{\boldsymbol{\psi}}_{i,k}(t)$ 和式(223)中的 $\hat{\boldsymbol{\Psi}}_{i,k}(L)$.

⑦ 根据式(238)选择 $\mu_{i,k}=2\lambda_{\max}^{-1}[\boldsymbol{\Psi}_{i,k}^{\mathrm{T}}(L)\cdot\boldsymbol{\Psi}_{i,k}(L)]$,用式(220)和(221)刷新参数估计向量 $\hat{\boldsymbol{d}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$.

⑧ 用(241)构造 $\hat{\boldsymbol{\theta}}_k$,用式(234)计算 $\hat{\boldsymbol{w}}_k(t)$,用式(237)计算 $\hat{\boldsymbol{v}}_k(t)$.

⑨ 比较 $\hat{\boldsymbol{c}}_k$ 和 $\hat{\boldsymbol{c}}_{k-1}$,比较 $\hat{\boldsymbol{d}}_k$ 和 $\hat{\boldsymbol{d}}_{k-1}$,及 $\hat{\boldsymbol{\theta}}_k$ 和 $\hat{\boldsymbol{\theta}}_{k-1}$,如果 $\|\hat{\boldsymbol{c}}_k-\hat{\boldsymbol{c}}_{k-1}\|\leqslant\varepsilon$, $\|\hat{\boldsymbol{d}}_k-\hat{\boldsymbol{d}}_{k-1}\|\leqslant\varepsilon$, $\|\hat{\boldsymbol{\theta}}_k-\hat{\boldsymbol{\theta}}_{k-1}\|\leqslant\varepsilon$,就结束计算,获得迭代次数 k 和参数估计 $\hat{\boldsymbol{c}}_k$, $\hat{\boldsymbol{d}}_k=\hat{\boldsymbol{d}}_{m,k}$ 和 $\hat{\boldsymbol{\theta}}_k$;否则,$k$ 增 1,转到步骤③.

4.5 基于滤波的部分耦合子系统最小二乘迭代辨识算法

极小化准则函数 $J_8(\boldsymbol{d},\boldsymbol{\theta}_i)$ 和 $J_9(\boldsymbol{c})$ 得到估计参数向量 $\boldsymbol{d}$, $\boldsymbol{\theta}_i$ 和 $\boldsymbol{c}$ 的最小二乘估计:

$$\begin{bmatrix}\hat{\boldsymbol{d}}_k\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\boldsymbol{\Psi}_i^{\mathrm{T}}(L)\boldsymbol{\Psi}_i(L)]^{-1}\boldsymbol{\Psi}_i^{\mathrm{T}}(L)\boldsymbol{Y}_{\mathrm{f}i}(L),\tag{242}$$

$$\hat{\boldsymbol{c}}_k=[\boldsymbol{Y}^{\mathrm{T}}(L)\boldsymbol{Y}(L)]^{-1}\boldsymbol{Y}^{\mathrm{T}}(L)\boldsymbol{W}_{\mathrm{n}}(L).\tag{243}$$

辨识的困难在于 $\boldsymbol{\Psi}_i(L)$ 中包含了不可测噪声项 $\boldsymbol{v}(t-j)$, $\boldsymbol{Y}(L)$ 中包含了不可测噪声项 $\boldsymbol{w}(t-j)$.多项式 $c(z)$ 是未知的,$\boldsymbol{Y}(L)$ 是未知的.输出矩阵 $\boldsymbol{W}_{\mathrm{n}}(L)$ 中包含了不可测项 $\boldsymbol{w}(t-j)$,因此计算参数估计 $\hat{\boldsymbol{d}}_k$, $\hat{\boldsymbol{\theta}}_{i,k}$ 和 $\hat{\boldsymbol{c}}_k$ 的算法无法实现.

采用辅助模型思想,将 $\boldsymbol{\Psi}_i(L)$ 用其迭代估计 $\hat{\boldsymbol{\Psi}}_{i,k}(L)$ 代替,$\boldsymbol{Y}(L)$ 用其迭代估计 $\hat{\boldsymbol{Y}}_k(L)$ 代替,$\boldsymbol{W}_{\mathrm{n}}(L)$ 用其迭代估计 $\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)$ 代替,能够获得辨识参数向量 $\boldsymbol{d}$, $\boldsymbol{\theta}_i$ 和 $\boldsymbol{c}$ 的基于滤波的部分耦合子系统最小二乘迭代算法(Filtering based Partially Coupled Subsystem Least Squares based Iterative algorithm, F-PC-S-LSI 算法):

$$\begin{bmatrix}\hat{\boldsymbol{d}}_{i,k}\\ \hat{\boldsymbol{\theta}}_{i,k}\end{bmatrix}=[\hat{\boldsymbol{\Psi}}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\Psi}}_{i,k}(L)]^{-1}\boldsymbol{\Psi}_{i,k}^{\mathrm{T}}(L)\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L),\tag{244}$$

$$\hat{\boldsymbol{c}}_k=[\hat{\boldsymbol{Y}}_k^{\mathrm{T}}(L)\hat{\boldsymbol{Y}}_k(L)]^{-1}\hat{\boldsymbol{Y}}_k^{\mathrm{T}}(L)\hat{\boldsymbol{W}}_{\mathrm{n},k}(L),\tag{245}$$

$$\hat{\boldsymbol{\Psi}}_{i,k}(L)=[\hat{\boldsymbol{\psi}}_{i,k}(1),\hat{\boldsymbol{\psi}}_{i,k}(2),\cdots,\hat{\boldsymbol{\psi}}_{i,k}(L)]^{\mathrm{T}},\tag{246}$$

$$\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)=[\hat{y}_{\mathrm{f}i,k}(1),\hat{y}_{\mathrm{f}i,k}(2),\cdots,\hat{y}_{\mathrm{f}i,k}(L)]^{\mathrm{T}},\tag{247}$$

$$\hat{\boldsymbol{Y}}_k(L)=\begin{bmatrix}\hat{\boldsymbol{\Omega}}_k(1)\\ \hat{\boldsymbol{\Omega}}_k(2)\\ \vdots\\ \hat{\boldsymbol{\Omega}}_k(L)\end{bmatrix},\quad \hat{\boldsymbol{W}}_{\mathrm{n},k}(L)=\begin{bmatrix}\hat{\boldsymbol{w}}_{\mathrm{n},k}(1)\\ \hat{\boldsymbol{w}}_{\mathrm{n},k}(2)\\ \vdots\\ \hat{\boldsymbol{w}}_{\mathrm{n},k}(L)\end{bmatrix},\tag{248}$$

$$\hat{\boldsymbol{\psi}}_{i,k}(t)=[\hat{\boldsymbol{\phi}}_{i,k}^{\mathrm{T}}(t),\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}^{\mathrm{T}}(t)]^{\mathrm{T}},\tag{249}$$

$$\hat{\boldsymbol{\phi}}_{k}(t)=[\hat{\boldsymbol{v}}_{k-1}(t-1),\hat{\boldsymbol{v}}_{k-1}(t-2),\cdots,\hat{\boldsymbol{v}}_{k-1}(t-n_d)]\tag{250}$$

$$=[\hat{\boldsymbol{\phi}}_{1,k}(t),\hat{\boldsymbol{\phi}}_{2,k}(t),\cdots,\hat{\boldsymbol{\phi}}_{m,k}(t)]^{\mathrm{T}},\quad(251)$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)=[-\hat{\boldsymbol{y}}^{\mathrm{T}}_{\mathrm{f},k}(t-1),-\hat{\boldsymbol{y}}^{\mathrm{T}}_{\mathrm{f},k}(t-2),\cdots,-\hat{\boldsymbol{y}}^{\mathrm{T}}_{\mathrm{f},k}(t-n_a),\ \boldsymbol{u}^{\mathrm{T}}_{\mathrm{f},k}(t-1),\boldsymbol{u}^{\mathrm{T}}_{\mathrm{f},k}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}_{\mathrm{f},k}(t-n_b)]^{\mathrm{T}},\quad(252)$$

$$\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)=\boldsymbol{y}(t)+\hat{c}_{1,k}\boldsymbol{y}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{y}(t-n_c)\quad(253)$$

$$=[\hat{y}_{\mathrm{f}1,k}(t),\hat{y}_{\mathrm{f}2,k}(t),\cdots,\hat{y}_{\mathrm{f}m,k}(t)]^{\mathrm{T}},\quad(254)$$

$$\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)=\boldsymbol{u}(t)+\hat{c}_{1,k}\boldsymbol{u}(t-1)+\cdots+\hat{c}_{n_c,k}\boldsymbol{u}(t-n_c),\quad(255)$$

$$\hat{\boldsymbol{\Omega}}_k(t)=[-\hat{\boldsymbol{w}}_{k-1}(t-1),\cdots,-\hat{\boldsymbol{w}}_{k-1}(t-n_c)],\quad(256)$$

$$\hat{\boldsymbol{w}}_k(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}_k\boldsymbol{\varphi}_{\mathrm{s}}(t),\quad(257)$$

$$\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}_{k-1}\boldsymbol{\varphi}_{\mathrm{s}}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_{k-1},\quad(258)$$

$$\boldsymbol{\varphi}_{\mathrm{s}}(t)=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),\ \boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}},\quad(259)$$

$$\hat{\boldsymbol{v}}_k(t)=\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)-\hat{\boldsymbol{\phi}}_k(t)\hat{\boldsymbol{d}}_k-\hat{\boldsymbol{\theta}}^{\mathrm{T}}_k\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t),\quad(260)$$

$$\hat{\boldsymbol{d}}_k=\frac{\hat{\boldsymbol{d}}_{1,k}+\hat{\boldsymbol{d}}_{2,k}+\cdots+\hat{\boldsymbol{d}}_{m,k}}{m},\quad(261)$$

$$\hat{\boldsymbol{c}}_k=[\hat{c}_{1,k},\hat{c}_{2,k},\cdots,\hat{c}_{n_c,k}]^{\mathrm{T}},\quad(262)$$

$$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}].\quad(263)$$

F-PC-S-LSI 算法(244)—(263)输出的参数估计为 $\hat{\boldsymbol{d}}_k=\hat{\boldsymbol{d}}_{m,k}$,$\hat{\boldsymbol{c}}_k=[\hat{c}_{1,k},\hat{c}_{2,k},\cdots,\hat{c}_{n_c,k}]^{\mathrm{T}}$,$\hat{\boldsymbol{\theta}}_k=[\hat{\boldsymbol{\theta}}_{1,k},\hat{\boldsymbol{\theta}}_{2,k},\cdots,\hat{\boldsymbol{\theta}}_{m,k}]$,其计算步骤如下:

① 确定数据长度 L,收集输入输出数据 $\{\boldsymbol{u}(t),\boldsymbol{y}(t):t=1,2,\cdots,L\}$,给定参数估计精度 ε,用式(259)构造信息向量 $\boldsymbol{\varphi}_{\mathrm{s}}(t)$.

② 置初值:令 $k=1$,$\hat{\boldsymbol{d}}_0=\mathbf{1}_{n_d}/p_0$,$\hat{\boldsymbol{\theta}}_0=\boldsymbol{I}_{n\times m}/p_0$,$\hat{\boldsymbol{y}}_{\mathrm{f},0}(t)$=随机向量,$\hat{\boldsymbol{u}}_{\mathrm{f},0}(t)=\mathbf{1}_m/p_0$,$\hat{\boldsymbol{v}}_0(t)$=随机向量,$\hat{\boldsymbol{w}}_0(t)$=随机向量.

③ 用式(256)构造 $\hat{\boldsymbol{\Omega}}_k(t)$,以构成式(248)中的 $\hat{\boldsymbol{Y}}_k(L)$;用式(258)计算 $\hat{\boldsymbol{w}}_{\mathrm{n},k}(t)$,以构成式(248)中的 $\hat{\boldsymbol{W}}_{\mathrm{n},k}(L)$.

④ 用式(245)刷新参数估计向量 $\hat{\boldsymbol{c}}_k$.

⑤ 从式(262)读出 $\hat{c}_{j,k}$,用式(253)计算 $\hat{\boldsymbol{y}}_{\mathrm{f},k}(t)$,用式(255)计算 $\hat{\boldsymbol{u}}_{\mathrm{f},k}(t)$,用式(252)构造 $\hat{\boldsymbol{\varphi}}_{\mathrm{f},k}(t)$;从(254)中读取 $\hat{y}_{\mathrm{f}i,k}(t)$,以构造式(247)中的 $\hat{\boldsymbol{Y}}_{\mathrm{f}i,k}(L)$.

⑥ 用式(250)构造 $\hat{\boldsymbol{\phi}}_k(t)$,并从(251)中读取 $\hat{\boldsymbol{\phi}}_{i,k}(t)$,以构成式(249)中的 $\hat{\boldsymbol{\psi}}_{i,k}(t)$ 和式(246)中的 $\hat{\boldsymbol{\Psi}}_{i,k}(L)$.

⑦ 用式(244)刷新参数估计向量 $\hat{\boldsymbol{d}}_{i,k}$ 和 $\hat{\boldsymbol{\theta}}_{i,k}$,用式(261)计算第 k 次的平均迭代估计值 $\hat{\boldsymbol{d}}_k$,用式(263)构成参数估计 $\hat{\boldsymbol{\theta}}_k$.

⑧ 用式(257)计算 $\hat{\boldsymbol{w}}_k(t)$,用式(260)计算 $\hat{\boldsymbol{v}}_k(t)$.

⑨ 比较 $\hat{\boldsymbol{c}}_k$ 和 $\hat{\boldsymbol{c}}_{k-1}$,比较 $\hat{\boldsymbol{d}}_k$ 和 $\hat{\boldsymbol{d}}_{k-1}$,及 $\hat{\boldsymbol{\theta}}_k$ 和 $\hat{\boldsymbol{\theta}}_{k-1}$,如果 $\|\hat{\boldsymbol{c}}_k-\hat{\boldsymbol{c}}_{k-1}\|\leqslant\varepsilon$,$\|\hat{\boldsymbol{d}}_k-\hat{\boldsymbol{d}}_{k-1}\|\leqslant\varepsilon$,$\|\hat{\boldsymbol{\theta}}_k-\hat{\boldsymbol{\theta}}_{k-1}\|\leqslant\varepsilon$,就结束计算,获得迭代次数 k 和参数估计 $\hat{\boldsymbol{c}}_k$,$\hat{\boldsymbol{d}}_k$ 和 $\hat{\boldsymbol{\theta}}_k$;否则,$k$ 增 1,转到步骤③.

5 结语

本文将耦合辨识概念与迭代搜索原理相结合,研究了多元受控自回归滑动平均系统和多元受控自回归自回归滑动平均系统的子系统梯度迭代辨识算法、部分耦合(子系统)梯度迭代辨识算法、子系统最小二乘迭代辨识算法、基于滤波的系统梯度迭代辨识算法、基于滤波的部分耦合(子系统)梯度迭代辨识算法和基于滤波的子系统最小二乘迭代辨识算法等.本文提出的算法可以推广到多元伪线性回归系统、类多变量方程误差类系统和类多变量输出误差类系统.

参考文献
References

[1] 丁锋.系统辨识新论[M].北京:科学出版社,2013
DING Feng. System identification: New theory and methods[M].Beijing:Science Press,2013

[2] 丁锋.系统辨识:辨识方法性能分析[M].北京:科学出版社,2014
DING Feng.System identification:Performance analysis for identification methods[M].Beijing:Science Press,2014

[3] 丁锋.系统辨识(1):辨识导引[J].南京信息工程大学学报:自然科学版,2011,3(1):1-22
DING Feng. System identification. Part A:Introduction to the identification[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(1):1-22

[4] 丁锋.系统辨识(2):系统描述的基本模型[J].南京信息工程大学学报:自然科学版,2011,3(2):97-117
DING Feng.System identification.Part B:Basic models for system description[J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition,2011,3(2):97-117

[5] 丁锋.系统辨识(3):辨识精度与辨识基本问题[J].南京信息工程大学学报:自然科学版,2011,3(3):193-226
DING Feng. System identification. Part C: Identification accuracy and basic problems[J].Journal of Nanjing University of Information Science and Technology: Natural Science Edition,2011,3(3):193-226

[6] 丁锋.系统辨识(4):辅助模型辨识思想与方法[J].南京信息工程大学学报:自然科学版,2011,3(4):289-318
DING Feng.System identification.Part D:Auxiliary model identification idea and methods[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(4):289-318
[7] 丁锋.系统辨识(5):迭代搜索原理与辨识方法[J].南京信息工程大学学报:自然科学版,2011,3(6):481-510
DING Feng.System identification.Part E:Iterative search principle and identification methods[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(6):481-510
[8] 丁锋.系统辨识(6):多新息辨识理论与方法[J].南京信息工程大学学报:自然科学版,2012,4(1):1-28
DING Feng.System identification.Part F:Multi-innovation identification theory and methods[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2012,4(1):1-28
[9] 丁锋.系统辨识(7):递阶辨识原理与方法[J].南京信息工程大学学报:自然科学版,2012,4(2):97-124
DING Feng.System identification.Part G:Hierarchical identification principle and methods[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2012,4(2):97-124
[10] 丁锋.系统辨识(8):耦合辨识概念与方法[J].南京信息工程大学学报:自然科学版,2012,4(3):193-212
DING Feng.System identification.Part H:Coupling identification concept and methods[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2012,4(3):193-212
[11] 丁锋.辨识方法的计算效率(1):递推算法[J].南京信息工程大学学报:自然科学版,2012,4(4):289-300
DING Feng.Computational efficiency of the identification methods.Part A:Recursive algorithms[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2012,4(4):289-300
[12] 丁锋.辨识方法的计算效率(2):迭代算法[J].南京信息工程大学学报:自然科学版,2012,4(5):385-401
DING Feng.Computational efficiency of the identification methods. Part B:Iterative algorithms[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2012,4(5):385-401
[13] 丁锋.辨识方法的计算效率(3):信息向量耦合算法[J].南京信息工程大学学报:自然科学版,2012,4(6):481-495
DING Feng.Computational efficiency of the identification methods. Part C:Coupled information vector algorithms[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2012,4(6):481-495.
[14] 丁锋,汪菲菲.多元系统耦合多新息随机梯度类辨识方法[J].南京信息工程大学学报:自然科学版,2014,6(1):1-16
DING Feng,WANG Feifei.Coupled multi-innovation stochastic gradient type identification methods for multivariate systems[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2014,6(1):1-16
[15] 丁锋,汪菲菲,汪学海.多元伪线性回归系统部分耦合多新息随机梯度类辨识方法[J].南京信息工程大学学报:自然科学版,2014,6(2):97-112
DING Feng,WANG Feifei,WANG Xuehai.Partially coupled multi-innovation stochastic gradient type identification methods for multivariate pseudo-linear regressive systems[J].Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2014,6(2):97-112
[16] 丁锋,汪菲菲,汪学海.类多变量输出误差系统的耦合多新息辨识方法[J].南京信息工程大学学报:自然科学版,2014,6(3):193-210
DING Feng, WANG Feifei, WANG Xuehai. Coupled multi-innovation identification methods for multivariable output-error-like systems[J]. Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2014,6(3):193-210
[17] Ding F,Liu G,Liu X P.Partially coupled stochastic gradient identification methods for non-uniformly sampled systems[J].IEEE Transactions on Automatic Control,2010,55(8):1976-1981
[18] Ding F.Coupled-least-squares identification for multivariable systems[J].IET Control Theory and Applications,2013,7(1):68-79
[19] Wang D Q,Ding F,Zhu D Q.Data filtering based least squares algorithms for multivariable CARAR-like systems[J].International Journal of Control,Automation,and Systems,2013,11(4):711-717
[20] Wang D Q, Ding F, Chu Y Y. Data filtering based recursive least squares algorithm for Hammerstein systems using the key-term separation principle[J].Information Sciences,2013,222:203-212
[21] Ding F,Chen T.Hierarchical gradient-based identification of multivariable discrete-time systems[J].Automatica,2005,41(2):315-325
[22] Ding F,Chen T.Hierarchical least squares identification methods for multivariable systems[J].IEEE Transactions on Automatic Control,2005,50(3):397-402

Partially coupled iterative identification methods for multivariable equation error type systems

DING Feng[1,2,3] WANG Feifei[1] WANG Xuehai[1]
1 School of Internet of Things Engineering, Jiangnan University, Wuxi 214122
2 Control Science and Engineering Research Center, Jiangnan University, Wuxi 214122
3 Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122

Abstract For multivariable equation error moving average systems, this paper gives an extended stochastic gradient identification algorithm, a recursive extended least squares identification algorithm, a gradient based iterative (GI) identification algorithm and a least squares based iterative (LSI) identification algorithm, using the least squares principle and the iterative search principle. For multivariable equation error moving average systems and multivariable equation error autoregressive moving average systems, this paper uses the coupling identification concept and decomposes a multivariable system into several subsystems, derives the corresponding GI identification algorithm, partially coupled (subsystem) GI identification algorithm, subsystem LSI identification algorithm and partially coupled subsystem LSI identification algorithm. Furthermore, using the filtering technique, this paper studies the subsystem GI identification algorithm, partially coupled (subsystem) GI identification algorithm, partially coupled subsystem LSI identification algorithm for multivariable equation error autoregressive moving average systems. The computational steps for several typical identification algorithms are provided.

Key words parameter estimation; iterative search principle; gradient search; least squares; data filtering technique; auxiliary model identification idea; hierarchical identification principle; coupling identification concept; multivariable system